Department of Precision and Microsystems Engineering

Planar Position Measurement of a Flat Object Based on Distributed Fiber Optic Sensors

Linghao Zeng

Report no   : 2024.085
Coaches   : Dr. ir. R.A.J. van Ostayen
                   : Dr. ir. J.F.L. Goosen
Specialisation   : Mechatronic System Design
Type of report   : Master Thesis
Date   : 26 September, 2024

TUDelft
Delft
University of
Technology

Abstract

This Master thesis describes the steps that have been taken to explore a new concept in the field of planar position measurement of flat and reflective objects using fiber optic sensors. The optical fibers are spread over the measurement plane, where transmitting fibers send out light profiles and the amount of light that is reflected from the object and detected by receiving fibers is utilized to decode the planar position of the object. The state-of-the-art distributed fiber optic sensor system shows potential for improvement in terms of accuracy, which is severely affected by the light interference from different light sources in the latest generation. To make use of this potential, the partial illumination scheme of transmitting fibers is adopted to improve the sensor system.
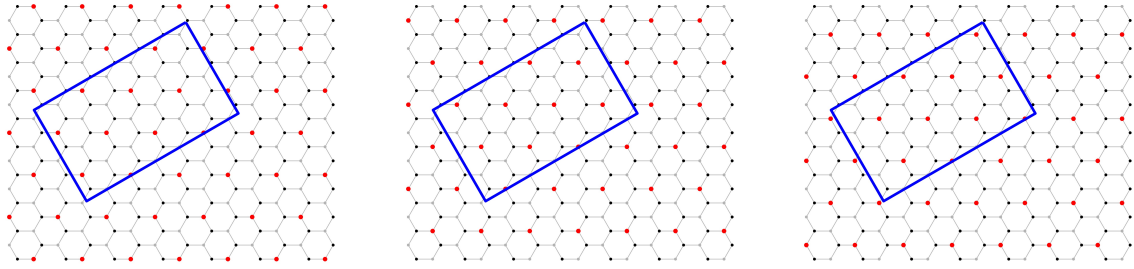

Figure 1: Partial illumination scheme of transmitting fibers in distributed fiber optic sensor system: Each receiving fiber denoted in black dot is surrounded by at most one illuminated transmitting fiber in the red dot for three types of illumination cases

The 2D Gaussian distributed intensity profile is applied to each illuminated transmitting fiber and the amount of light collected by a receiving fiber is calculated by a surface integral over light intensities under the region of the measured object. The theoretical brightness indicated by these numerical values is approximated by a multivariate function concerning the distance and orientation of the edge adjacent to a receiving fiber as well as the fiber density. Once the orientation is decoded by a fiber array with a fixed fiber density, the predicted distance can be solved by the inverse function of the fitting model.

To validate the theoretical model, the brightness of a receiving fiber dependent on edge distance is calibrated by a prototype with a single set of transmitting and receiving fibers. The developed prototype enables a relative planar positioning between the reflective surface and the receiving fibers, with a translation resolution of up to 0.05 mm. The type of diffuser that greatly expands the light spot and detection scope of the fibers is first determined by this prototype. Afterward, the achievable resolution of brightness curves is evaluated for various distances between transmitting and receiving fibers. As a result, the fiber distance that corresponds to an 8 mm in-radius of hexagons in the sensor array is chosen. The resolution of this configuration reaches 0.1 mm with a relatively sparse fiber distribution. The diffusive radius of the receiving fiber is expanded to 7.2 mm and the goodness of fit for the brightness curve reaches 97%.

In the next phase, the algorithms that derive the planar DoFs of an object are developed on the fiber array with a determined fiber distance. The orientation of the object is initially estimated by the positions of receiving fibers with sufficient brightness values, then it is applied to predict the distance between each receiving fiber and the boundary of an object based on the explicit expression of the fitting model. Afterward, the preliminary estimate is refined by searching for a planar position with a minimum sum of square differences between the geometric and estimated distances for all receiving fibers, this process is repeated until the orientation is stabilized. Among 60 test samples at random positions, the position error is less than 0.15 mm with a 90% confidence level, and the median angular error is limited to 0.1 milli-rad.

The prototype of a distributed fiber optic sensor system has been designed and manufactured. Each light source aligns with transmitting fibers in the same row of the array, resulting in six LED lamps for each group of transmitting fibers. Different groups of light sources are independently controlled by digital pins of the Ar-

duino microcontroller. These electronic elements are placed at the bottom layer of the prototype. The sensor surface that holds the transmitting and receiving fibers is located at the upper layer, where the other end of these fibers are aligned to light sources and a camera via designed collimators.
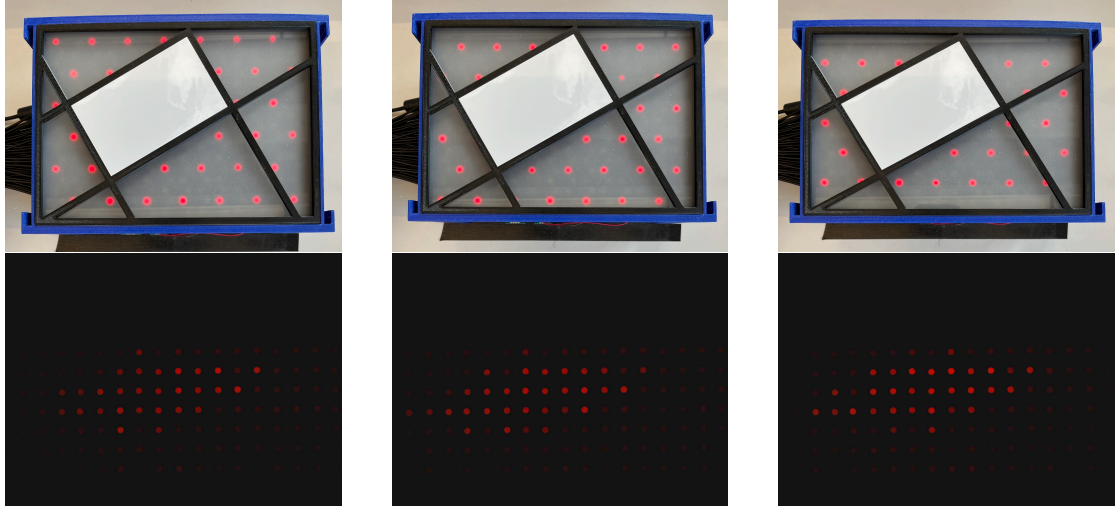

Figure 2: Prototype layout and related image inputs in different illumination cases used for position measurement

Figure 2 shows the distributed fiber optic sensor system layout and image inputs in different illumination cases. The light spots are transformed into brightness and positions of corresponding receiving fibers on the sensor surface. These data are processed by the established planar positioning algorithms that return the planar DoFs of the measured object. For 20 position samples, this sensor configuration demonstrates 0.77 mm and 0.65 mm median error along horizontal and vertical directions, with angular deviation less than 1°. The difference in error between simulation and experiments can be attributed to the brightness loss of the light profile, which causes a shift of estimated edge position in the fitting model.

To conclude, the distributed fiber optic sensor system with a partial illumination scheme manages to limit the overall planar position error to the sub-millimeter scale, while maintaining an angular deviation of less than 1 degree. The measurement range is greatly extended and the sensing element is no longer required to be attached to the measured object. Therefore, this sensing configuration is promising for planar position measurement, especially for thin and fragile products with specular surfaces.

Contents

Nomenclature

This section lists the names and definitions of all abbreviations and symbols used in the main body of the thesis. Subscript characters in regular font within the symbols are used solely to distinguish this variable from others and do not have any physical meaning. Symbols with similar definitions are grouped using a subscript set enclosed in curly brackets.

Abbreviations

Abbreviation	Definition
AWG	American Wire Gauge
CAGR	Compound Annual Growth Rate
CCD	Charge Coupled Device
DC	Direct Current
DoF(s)	Degree(s) of Freedom
FDM	Fused Decomposition Modelling
LED(s)	Light Emitting Diode(s)
MSLA	Masked Stereo Lithography Apparatus
PCB	Printed Circuit Board
PLA	Polylactic Acid
PMMA	Polymethyl Methacrylate
PV	Photovoltaics

Symbols

Symbol	Definition	Unit
d, d_{pq}	Geometric distance from the receiving fiber to the edge	mm
$\widehat{d}, \widehat{d}_{pqt}$	Estimated distance from the receiving fiber to the edge by the inverse fitting model N_{F}^{-1}	mm
$\widehat{d}_{\mathrm{l}}, \widehat{d}_{\mathrm{u}}$	Critical distances when the detection scope of fiber is fully exposed/covered with minimum/maximum output N	mm
$E_p\{(x_{pq}, y_{pq})\}$	Coordinate sets of receiving fibers adjacent to different edges	-
F	Shifted Sigmoid model between N_{F} and d	-
$f_{\mathrm{d}}(x, y, \alpha)$	Square sum of differences between geometric distance d and estimated distance $\widehat{d}$ when object is located at (x, y, α), objective function of optimization	mm^2
$G_{\{1,2\}}$	Coefficients in the theoretical expression of N that are dependent on θ	-
G_{R}	Group number of a receiving fiber	-
G_{T}	Group number of a transmitting fiber	-
$g_{\mathrm{l}}(x, y, \alpha)$	Difference $\widehat{d}_l - d$, constraints of optimization with non-positive values	mm
$g_{\mathrm{u}}(x, y, \alpha)$	Difference $d - \widehat{d}_u$, constraints of optimization with non-positive values	mm
$H_{\{1,2,3\}}$	Coefficients in N_{F} related to sensor surface parameters r_{c}/r_0	-
h	Length of the measured rectangular object	mm
$I_{\{1,2\}}$	Branch currents flowing through the potentiometer	mA
I_{b}	Base current of transistor	mA
I_{c}	Collector current flowing through the light source	mA
$I_{\mathrm{c,max}}$	Maximum forward current in LED	mA
$I_{\mathrm{c,nom}}$	Nominated current in LED	mA
$I(x, y)$	Light intensity at certain planar position	W/m^2

Symbol	Definition	Unit
$J_n(x_B, y_B)$	The maximum projection length from (x_B, y_B) onto the line formed by two vertices included in coordinate set E_n	mm
$\widehat{l_n}(x_B, y_B)$	Distance from (x_B, y_B) to the line formed by two vertices included in coordinate set E_n	mm
l_p	Implicit linear edge function of edges in the measured rectangular object	-
m, n	Indices used to distinguish different edges of the rectangular boundary	-
N	Normalized light power	-
N_E	Empirical expression of normalized light power	-
N_F	Fitting model for normalized light power by basic elementary functions	-
P	Power of light detected by a receiving fiber, surface integral of light intensity over the effective reflection zone	W
$\widehat{P}$	Averaged R value of pixels within effective region $\hat{S}_R$ for a receiving fiber	-
P_U, P_L	Maximum/Minimum theoretical light power detected by a receiving fiber	W
$\widehat{P_U}, \widehat{P_L}$	Maximum/Minimum averaged R value for a receiving fiber	-
p	Index of edge closest to a receiving fiber	-
q	Index used to distinguish different fiber positions in a set E_p	-
R	Variable resistance in the potentiometer	Ω
$R(u, v)$	R value of a pixel located at column u and row v in an image	-
R^2	Coefficient of determination	-
R_b	Resistance of the resistor at the base of transistor	Ω
R_p	Total resistance of potentiometer	kΩ
r_0	Radius of light spot emitted from a fiber	mm
r_c	Radius of inscribed circle of the hexagon cell in the sensor array	mm
r_R	Radius of the detected receiving fiber in a snapshot	-
S	Plane region of the effective reflection zone $S_O \cap S_R$	mm^2
$S_B\{(x_B, y_B)\}$	Set of coordinates of receiving fibers near the object's boundary	-
S_O	Plane region enclosed by the boundary of the measured rectangular object	mm^2
S_R	Plane region of the detection scope by a receiving fiber	mm^2
$\hat{S}_R$	Set of effective pixels to be averaged for a receiving fiber in an image	-
S_T	Plane region of the light spot emitted from a transmitting fiber	mm^2
$S_v\{(x_{v_i}, y_{v_i})\}$	Set of coordinates of receiving fibers near the object's vertices	-
$S_{\{x_U, x_L, y_U, y_L\}}$	Candidate position sets for each corner when corner coordinates take on extreme values at either x or y directions	-
$S_{\{x_U y_U, ..., x_U y_L\}}$	Candidate position sets for each corner when corner coordinates take on extreme values at both x and y directions	-
t	Position index of a transmitting fiber relative to an adjacent receiving fiber	-
U	Column index of a receiving fiber from left to right in the camera frame	-
U_R	Row index of a receiving fiber from top to bottom in the sensor array	-
U_T	Row index of a transmitting fiber from top to bottom in the sensor array	-
(u, v)	Column and row number of a pixel in an image	-
(u_R, v_R)	Center position of a circular light spot in an image	-
V	Row index of a receiving fiber from top to bottom in the camera frame	-
V_B	Electric potential at the wiper of potentiometer	V
V_{be}	Threshold voltage of the conducting base-emitter junction in transistor	V
V_{cc}	Nominated voltage of digital output pins from Arduino Uno	V
V_{sup}	Power supply for LED arrays in the sensor surface application	V
w	Width of the measured rectangular object	mm
X_L, X_U, Y_L, Y_U	Mapping from position set to subset with maximum(U)/minimum(L) value in $x(X)/y(Y)$ direction	-
(x_0, y_0)	Geometric center of the measured rectangular object	mm
(x_a, y_a)	Intersection point between light profile and detection scope with the minimum fiber output	mm
(x_b, y_b)	Intersection point between light profile and detection scope with the maximum fiber output	mm

Symbol	Definition	Unit
$(x_{\text{c}}, y_{\text{c}})$	Final solution of center position of the measured rectangular object	mm
(x_p, y_p)	Vertex coordinates of the measured rectangular object	mm
$(x_{\text{R}}, y_{\text{R}})$	Global coordinate of a receiving fiber	mm
$(x_{\text{T}}, y_{\text{T}})$	Global coordinate of a transmitting fiber	mm
α	Angle between the principal axis of the measured rectangular object and the horizontal axis	rad
α_0	Initial estimation of object angle α based on fiber coordinates	rad
α_{c}	Final solution of object angle α	rad
α_p	Angle swept counterclockwise from the vertical axis in the global coordinate to the edge directed from outside to inside of the rectangular boundary	rad
$\widehat{\alpha_p}$	Estimated α_p derived by the slope of the regression line related to coordinate set E_p	rad
$\alpha_{x'}$	Estimated rotation angle of the object coordinate $x'Oy'$ with respect to the global coordinate xOy in the process of corner recognition	rad
β, β_{cs}	DC current gain at collector currents of 50/100 mA	-
θ	Angle swept counterclockwise from the vertical axis in the local coordinate of the fiber cell to the edge directed from outside to inside of the rectangular boundary	rad
σ	Standard deviation of theoretical Gaussian light profile	-

1

Introduction

1.1. Motivation

1.1.1. Ultra-thinned wafer

With the technology breakthrough in wafer thinning, there is a growing trend among industries to develop semiconductor devices with reduced packaging size, better electronic performance, and more sensitive structures. Today, most semiconductor wafers are thinned down to a range of 100 μm to 200 μm especially when it comes to power applications, solar cells, and image sensors. The demand for these products will continue to increase with an overall compound annual growth above 5%, as shown in Figure 1.1.
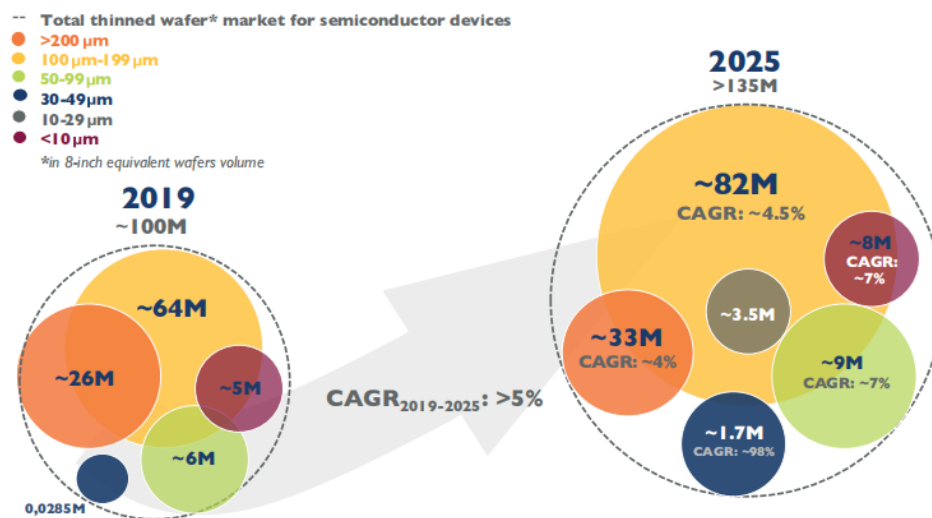
Figure 1.1: Thinned wafer market volume: 2019-2025 breakdown by thickness range[2]

Among the semiconductor industries, photovoltaics is a fast-growing market with about 26% CAGR of cumulative PV installation from 2013 to 2023[41]. The prime technology for the photovoltaic industry is dominated by crystalline silicon wafer(c-Si), with over 95% of the total production in 2023 from Figure 1.2. Compared with other technologies that have phased out in recent years, the silicon wafer in mono-crystalline solar cells yields the highest energy conversion efficiency and the wafer thickness has reduced to around 150 μm to absorb more incident sunlight and collect photo-excited current with less energy loss.

Figure 1.2: Percentage of global annual PV production[41]

In addition to thickness reduction, manufacturers are also focused on producing larger c-Si wafers for solar panels as the wafer with a larger surface area generates more power. In the early 2000s, solar wafers were small, around 125 m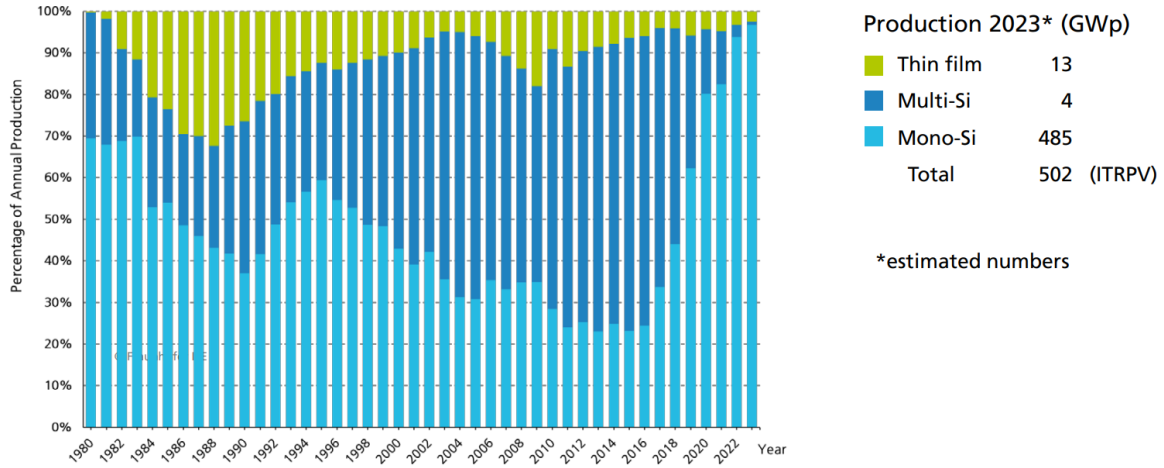m in length. In 2012, the M0 wafer (156 mm) was introduced with the M2 wafer (156.75 mm) eventually becoming the dominant size. The last few years have seen increased interest in even larger wafers with M6 (166 mm), M10 (182 mm), and G12/M12 (210 mm). In 2019, the solar panel supplier Sun-Power jumped from 125-mm wafers used in its *X-series* to M4-equivalent wafers (161.75 mm) for its *A-series* of modules, see Figure 1.3. The new *Performance 5 panels* will use even larger wafers and SunPower has made a large effort to drive down installation times and make G12 modules adaptable to the residential market.

Figure 1.3: Wafer size renovation by solar cell supplier[42]

1.1.2. Contactless positioning system

From the point of view of device performance, the wafer should be typically thinned with less than 200 μm, while the mechanical stability of such ultra-thin wafers would be greatly reduced especially for bigger wafer sizes. Microcracks due to warp and twist influenced by gravity could be a more prevalent problem during the manufacturing process in full-size wafers larger than M6.

One way to overcome the mentioned problem is to levitate the fragile product by contactless handling systems. Existing research has come up with methods to actuate the thin and flat substrates by aerostatic or electromagnetic forces where high-resolution actuators with large strokes are equipped[52, 65], in this case, sensors used to detect the object and control the actuators must be able to measure with sufficient precision and resolution in long range.

Typically, the vision-based measurement approach by CCD camera is one of the most used solutions. In this sensor configuration, the CCD camera is mounted over the operational platform that covers the object's

whole motion range in its visual scope. When the workspace becomes larger, the expanded vision field leads to deterioration in measurement resolution. In addition, other obstructions, such as manufacturing devices over the handled object, may block the view of the CCD camera and result in detection interference. Thus, developing a suitable position sensing scheme for ultra-thin wafers in solar panels is the main driver of this research, as sensor characteristics directly determine the overall positioning performance of the system such as repeatability, accuracy, and resolution. Non-tactile positioning sensors viewed from the bottom by different principles have been investigated with an overview given as follows.

Capacitive sensor is composed of a fixed surface with periodically positioned excitation electrodes. The conductive mover substrate above the electrode array forms a variable capacitor, with a constant capacitance change per distance. The capacitive sensor can provide micrometer scale resolution for displacement detection[6, 24, 40, 58, 59], while the dielectric constant of the electrode plate is sensitive to environmental variations such as temperature, humidity, air pressure, so they are rarely used in harsh environments with strong electromagnetic field or disturbance of airflow.

Magnetic sensor uses a Hall element that outputs a voltage in proportion to the magnetic field normal to the surface of the element. For non-ferromagnetic material detection, the mover substrates are suspended above the magnetic array with a magnetic field that is expressed explicitly in terms of in-plane locations[8, 16, 35, 62]. Hall sensors are installed on the moving stages to decode the in-plane position from the magnitude of the magnetic field at sensor positions. This poses a problem since sensor attachment on fragile objects is undesirable.

Optical mouse sensor captures the diffusive light pattern reflected from the object surface. The movement of local light patterns in consecutive frames detected by the photo-sensor can be used to derive the in-plane displacement[1, 37]. By optical sensing medium, no extra element is required to be mounted on the measured object. The problem arises when detecting solar panels since the uniform light reflected by specular surfaces makes it difficult to track the displacement.

Fiber optic sensor uses transceiver fiber bundles to detect the objects location. The optical fiber consists of two layers with distinct reflecting indexes through which the light beam travels with little loss. Each transmitting fiber sends out a light profile over the sensor surface while receiving fibers collect reflected light beams from the object surface with outputs of light intensity read by the photo-sensor. The requirement for the measured object for this sensing method is a flat mirror surface, which is the case for ultra-thin wafers used for solar panels. This approach offers new possibilities with many advantages: the measurement of large strokes is enabled by the arrangement of many small inexpensive fibers. It is also highly robust to failures due to the redundancy of the mechanism. By comparison, this technology will be pushed forward in this research.
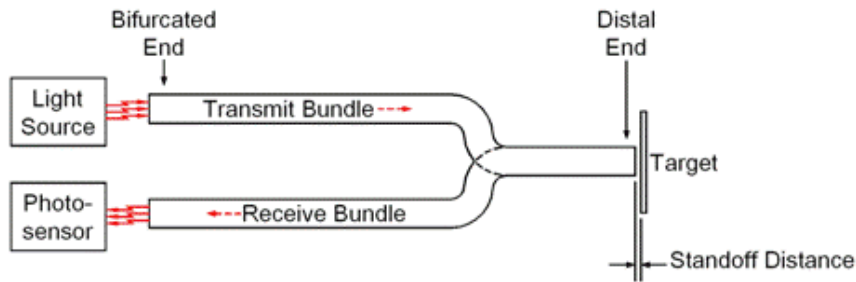
Figure 1.4: Fiber optic sensor layout[13]

1.2. State of the art

The first experiment on optical fibers not for telecommunications, but for sensor purposes, can be originated from the early 1970s. The field has continued to progress since that time and two major types of fiber optic sensors, interferometry-based and intensity-based sensors, have been developed. The intensity-based fiber optic sensors are the earliest and most widely used technology to date because of its low cost, easy installation, and high sensitivity[18, 55]. These sensors are conventionally applied for single-axis measurement as the amount of light collected by the bundle fiber is directly correlated to the displacement between the fiber and

the reflective surface. In recent years, Extensive studies have put more effort into applying fiber optic sensors for higher DoFs such as in-plane dimensions with innovations in fiber configurations, light modulation, and intensity decoding methods. These results as well as underlying concerns about sensor application will be discussed in this section.

1.2.1. Patent outcomes

The planar positioning fiber optic sensors from discovered patents can be mainly classified into two types: gap distance variant and reflection zone variant sensing schemes. The schematic view of the fiber layout can be found in Figure 1.5. According to the figure on the left, light pattern change dependent on the gap distance between fiber tips and the object surface can be utilized to measure the spatial motion of the target plane. Varied gap distance was formed by the tilted target surface(210) relative to the fiber array(220). In this case, distinct ring-shaped light patterns can be projected on the image sensor(230) from different fiber bundles. When the target surface moved relative to the optical path array, imaging sensors captured the shift of ring-shaped patterns to evaluate displacement along in-plane dimensions[28].
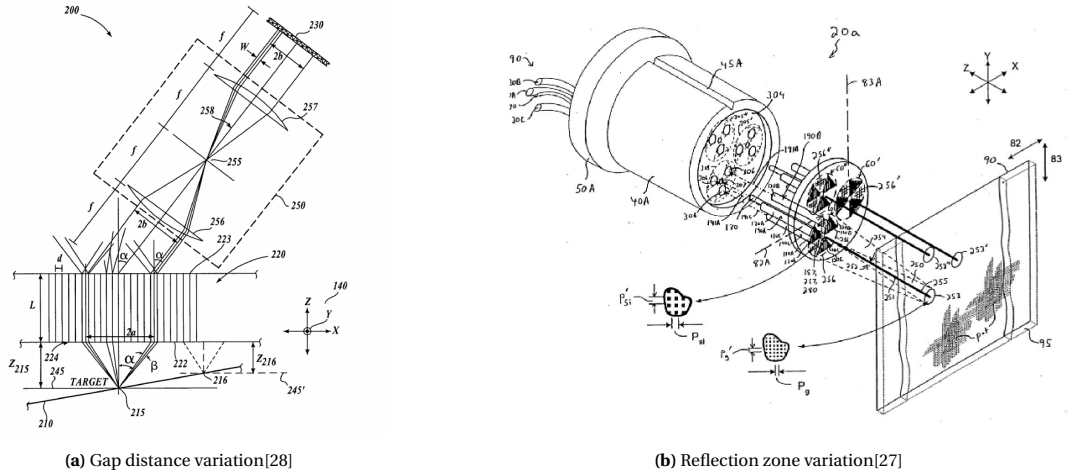

(a) Gap distance variation[28]   **(b)** Reflection zone variation[27]

Figure 1.5: In-plane positioning patents by fiber optic sensors

The greatest challenge of the former sensing scheme lies in the surface inclination, especially for large and fragile objects, and this could be avoided in the latter approach. According to Figure 1.5b, a fixed fly height was set between fiber optic sensing probes(40A) and the measured object(95). Three sets of receiving fibers in the sensing probe were covered by grating masks(256, 256, 256) in horizontal or vertical directions. A planar grating scale(90) in micrometer resolution was covered on the surface of the object, with various grating scales along the X and Y directions[27]. The relative motion between the grated planar object and the masks on the probe causes a periodically changed detection scope of the receiving fibers, thereby generating an intensity curve in the same period as the variation of the detection scope. The in-plane displacement can be determined by the number of occurred intensity cycles detected by different receiving fibers, as each cycle was directly related to the grating size.

1.2.2. Scientific researches

The research groups at TU Delft have been focusing on developing a smart surface that can detect thin substrates levitated by air film from the bottom side of the bearing table, where fiber optic sensors were placed in top-priority in design. The initial smart surface platform spreads the fiber transceiver sets over the measurement plane, see Figure 1.6. This arrangement greatly expanded the measurement scope without attaching additional elements to the measured object compared with concentrated fiber probes used in patents. In the illustrated prototype setup, the photosensor beneath the measurement plane determines if the object is placed above the receiving fibers based on the brightness of the light spot from corresponding fiber tips. The binary object states of discrete fiber positions determined by an intensity threshold led to a millimeter-level accuracy for planar positioning[12].
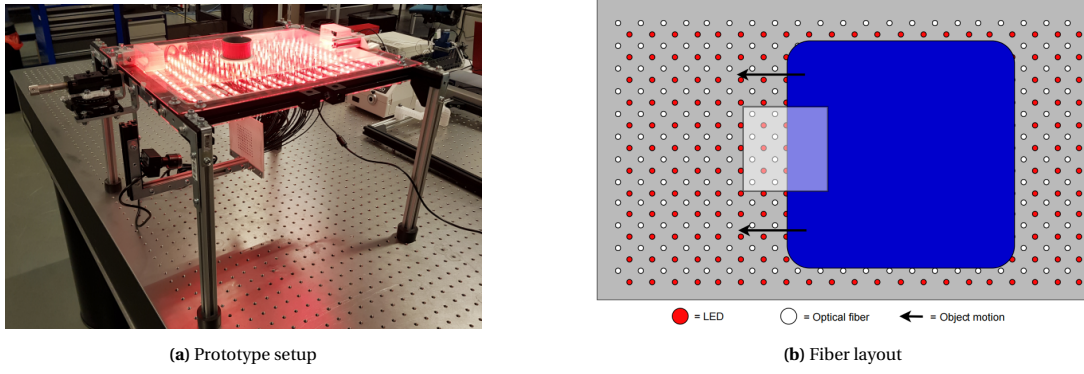
(a) Prototype setup

(b) Fiber layout

Figure 1.6: Smart surface concept by fiber optic sensors with binary intensity levels[12]

The distributed fiber optic sensor system was improved in the subsequent research that aims to make the most of light intensity levels and this attempt was carried out in a single set of fiber transceivers among the sensor array. A volumetric light diffuser was introduced between the measured object and the optical fibers and it was proved to effectively expand the radius of the light spot as well as the detection scope of optical fibers by a prototype experiment, with smooth intensity transitions from bright to dark patterns[4]. In the third generation of smart surface design, the receiving fiber was surrounded by 3 groups of transmitting fibers that emitted distinct primary colors in a hexagonal array. The combination of colors of light within the detection scope could indicate the proximity of the object to the specific light source[36].
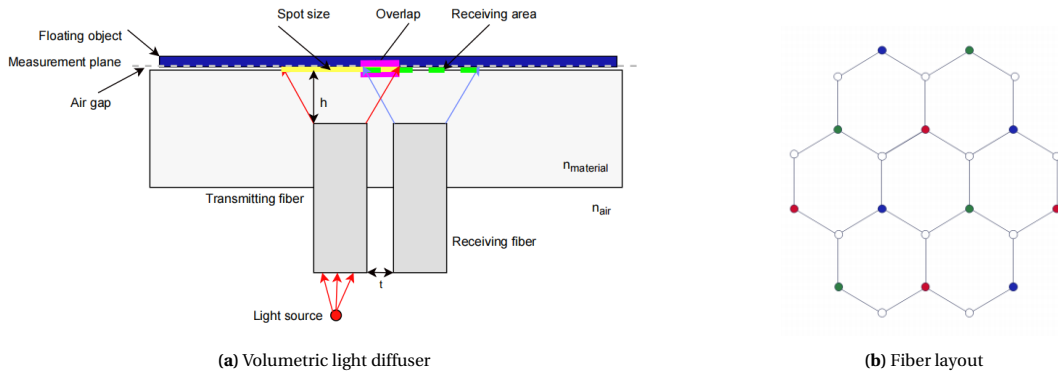

(a) Volumetric light diffuser

(b) Fiber layout

Figure 1.7: Smart surface concept by fiber optic sensors with multiple intensity levels[4, 36]

1.3. Research objective

Among the latest studies on fiber optic sensors mentioned in the previous section, there is still a huge step from the proposed conceptual design to planar positioning applications for fiber optic sensors. Although previous literature concepts have proved to expand the measurement range of the fiber optic sensors without additional elements attached to the measured object, there are still some shortcomings in the current solutions. The sensing scheme still has significant space for improvement in accuracy. For instance, the light interference from different sources as well as distinct sensitivity to light in different colors in the latest generation of fiber optic sensors made it difficult to quantify the relationship between light intensity and in-plane degrees of freedom. An alternative solution to this fiber configuration is to illuminate each group of transmitting fiber in the same color by a time sequence so that each receiving fiber is surrounded by at most one light source. The retained fiber division strategy offers possibilities for accuracy improvement while the density of illuminated fibers is reduced to avoid light interference. Furthermore, the position decoding method will be explored by distributed fiber optic sensors with multiple intensity levels to check the feasibility of the sensor application. Since these concepts have not been done so far, the objective of this research is stated as:

"Theoretical and experimental determination of the accuracy of a distributed fiber optic sensor system intended to measure the planar position of thin substrates, and derivation of optimal design parameters for this sensor

system."

1.4. Report outline

The remainder of this report focuses on the modeling, design, manufacture, and validation of the distributed fiber optic sensor system. The detailed fiber configurations and the theoretical relationship between fiber brightness and object position are explained in Chapter 2. The next chapter validates the relationship by a prototype containing fibers in a unit cell and determines the geometric parameters of the sensor array. The simulated fiber optic sensor array is built up and the planar positioning algorithms based on this platform are proposed in Chapter 4. The next chapter discusses the manufacturing procedures of the distributed fiber optic sensor system. Finally, the developed algorithms from simulations are applied to the prototype of the sensor array and the results are compared with the theoretical model.

2

Concept exploration

The general working principle of fiber optic sensors discussed in the introduction will be explored in this chapter from theoretical analysis. This starts with a simulated fiber optic sensor array, where the overall arrangement of fiber transceivers is explained to introduce the design parameters of the sensor array. The theoretical sensor outputs dependent on the planar positions of the measured object and sensor surface parameters are derived from Gaussian distributed light intensity patterns emitted from transmitting fibers, and mathematical fitting models in elementary functions are built up to interpret their relations.

2.1. Fiber array configuration

The type of grid where the optical fibers are placed is considered to be one of the key factors in the precision of the sensing system. Elfferich et al. carried out simulations in their research to evaluate the optimal fiber distribution patterns for detecting wafer substrates. The hexagonal grid was indicated to be a promising configuration compared with other types such as uniform triangles or squares for a lower positioning error when averaging across all fiber densities, see Figure 2.1a. This hexagonal array was also applied in the actuator design of the contactless handling system for thin substrates. Due to the great potential revealed by theoretical analysis and prototype development in previous studies, the sensor surface with hexagonally distributed fibers will be developed in this research.
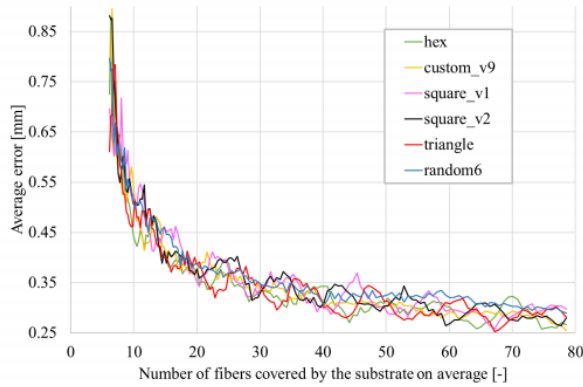
(a) Position error of grid patterns evaluated at different fiber density[12]
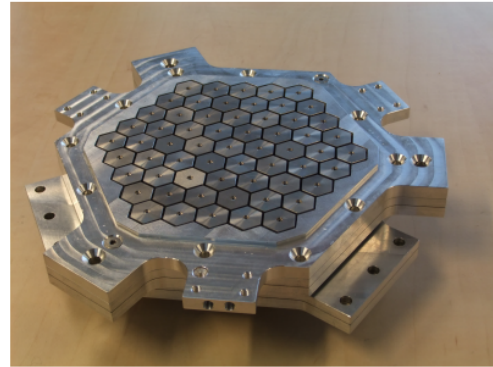
(b) Flowerbed air-bearing table[52]

Figure 2.1: Application of hexagonal array in thin substrate positioning platform

Figure 2.2 illustrates the proposed fiber configuration for the sensor surface. A rectangular mirror block in the same shape as the solar panel but with a smaller size is selected as the measured object, which is illustrated in blue lines. The geometric information of the object can be employed in position measurement as substrates are produced in fixed dimensions. The sensor array is first built within a standard A5-sized area

(210×148 mm^2). All the fibers within this area are considered as a single unit, and the fiber optic sensor surface can be expanded by this unit in future research to measure full-sized wafers used for solar panels.

The receiving fibers denoted in black dots are uniformly distributed over the hexagonal array, with at most three adjacent transmitting fibers for each element. According to the concept of partial illumination proposed in the research objective, the transmitting fibers in the array are divided into three groups, with each group being illuminated sequentially. The illuminated transmitting fibers in each group are highlighted in red dots.
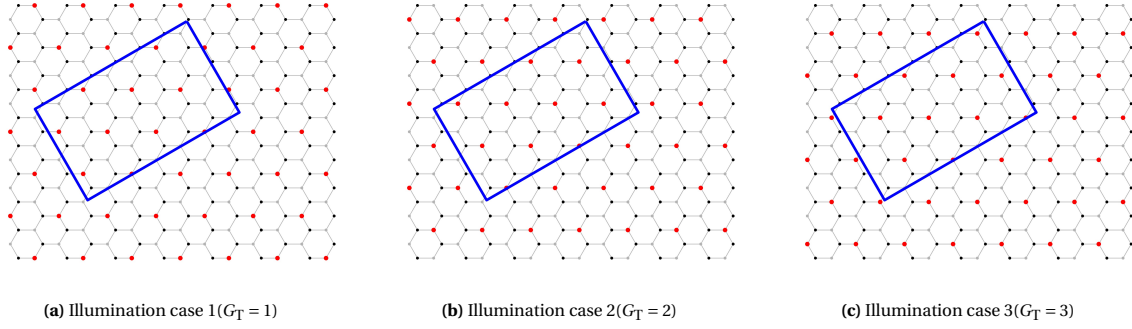

<table>
<tr><td>(a) Illumination case 1($G_T = 1$)</td><td>(b) Illumination case 2($G_T = 2$)</td><td>(c) Illumination case 3($G_T = 3$)</td></tr>
</table>

Figure 2.2: Partial illumination by different groups of transmitting fibers in sensing surface

For both transmitting and receiving fibers, their group numbers G_T and G_R are determined by their respective row number U_T and U_R in the array from top to bottom. These group numbers are expressed as the following equations and will be used to determine the relative position between the transmitting fiber and the receiving fiber.

$$G_T = \mathrm{mod}(U_T - 1, 3) + 1 \tag{2.1}$$

$$G_R = \mathrm{mod}(U_R - 1, 3) + 1 \tag{2.2}$$

The major parameters for fiber optic sensor surface are described as follows.

Fiber probe density is defined as the number of receiving fibers that are used to detect the proximity of the object in a given range. As the fibers are arranged regularly and repeatedly, the radius of the inscribed circle of the hexagonal cell in the sensor array, denoted as r_c, could be a more intuitive way to evaluate the density by fiber intervals. More receiving fibers can be accommodated on the sensor surface with smaller hexagonal cells to improve accuracy, while this is accompanied by greater complexity for fiber installation and data processing. This contradiction leads to an optimal value for fiber probe density that will be figured out by numerical method as well as prototype experiment in the subsequent chapters.

Fiber diffusivity can be defined as either the maximum region on the measurement plane that the light emitted from a transmitting fiber can reach or the region from which the reflected light can be detected by a receiving fiber. This equivalence is attributed to the reversibility of optical paths. A larger detection scope of receiving fibers indicates greater sensitivity in position measurement. Fiber diffusivity is dependent on diffuser performance, making it difficult to manually regulate the size. However, this parameter could also be reflected in the transition range of the intensity curve from light to dark pattern when the measured object crosses over the receiving fiber. An experiment will be conducted to determine the maximum reachable diffusivity of a single set of fiber transceivers.

2.2. Light propagation model

The theoretical light intensity collected by receiving fibers in the sensor surface affected by object position will be described in this section. The derivation originates from a unit cell of the sensor array, with one receiving fiber surrounded by three transmitting fibers as illustrated in the colored region in Figure 2.3. According to the partial illumination schemes proposed in Figure 2.2, the position of illuminated transmitting fiber (x_T, y_T)

and its index t adjacent to certain receiving fiber at (x_R, y_R) are defined as

$$
(x_T, y_T) = \begin{cases} (x_R + \dfrac{2r_c}{\sqrt{3}}, y_R) & t = 1 \\[2mm] (x_R - \dfrac{r_c}{\sqrt{3}}, y_R + r_c) & t = 2 \\[2mm] (x_R - \dfrac{r_c}{\sqrt{3}}, y_R - r_c) & t = 3 \end{cases}
\tag{2.3}
$$

The position index t is determined by the group number of transmitting fiber G_T and receiving fiber G_R, the mapping is denoted as T_G and it is expressed as:

$$
t = T_G(G_T, G_R) = \begin{cases} 1 & G_T = G_R \\ 2 & \mathrm{mod}(G_T, 3) + 1 = \mathrm{mod}(G_R, 3) \\ 3 & \mathrm{mod}(G_R, 3) + 1 = \mathrm{mod}(G_T, 3) \end{cases}
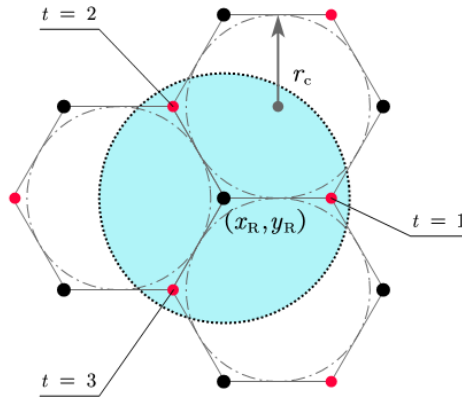\tag{2.4}
$$

Figure 2.3: Arrangement of fiber transceivers in a unit cell

2.2.1. Light intensity profile

The light emitted from a broad-band light source forms a spot on the surface of the measured object through the plastic optical fiber, with a Gaussian distributed intensity profile in two dimensions[26, 36, 61] expressed by Equation 2.5.

$$
I(x, y) = \begin{cases} C_0 \exp\left(-\dfrac{(x - x_T)^2 + (y - y_T)^2}{2(r_0\sigma)^2} \right) & (x, y) \in S_T \\[2mm] 0 & (x, y) \notin S_T \end{cases}
\tag{2.5}
$$

$$
S_T = \{(x, y) \in \mathbb{R}^2 \mid (x - x_T)^2 + (y - y_T)^2 \le r_0^2\}
\tag{2.6}
$$

In the above equation, C_0 is a constant with an intensity dimension of [W/m^2]. The light profile is centered on the central position of the corresponding transmitting fiber (x_T, y_T) within the fiber diffusivity range S_T. The diffusive light region can be bounded by a radius r_0 due to circular isolines in 2D Gaussian distribution functions. σ is the standard deviation of the Gaussian profile that determines the amplitude distribution. Since the light intensity at the edge of the light spot is almost negligible compared to the light intensity at its center, which differs by approximately two orders of magnitude, the value of σ can be derived, with a result of 0.33 calculated using Equation 2.7.

$$
\lim_{(x - x_T)^2 + (y - y_T)^2 \to r_0^{2-}} I(x, y) = 10^{-2} I(x_T, y_T)
\tag{2.7}
$$

2.2.2. Effective light reflection zone

The propagated light profiles are partly reflected by the object and collected by the receiving fibers on the sensor surface. The mathematical expression of the enclosed region by rectangular object S_O is required to be derived, as this constrains the detection scope from which the emitted light from the transmitting fibers can be reflected. The parameter definition of the measured object centered at (x_0, y_0) is illustrated in Figure 2.4. The orientation of the object α is defined as the angle between its main axis along the length direction x' and the horizontal axis x. According to the symmetric properties of the rectangle, this angle ranges from 0 to 180 degrees.

The absolute angle of the edge α_p, illustrated by red arcs in Figure 2.4, is defined as the angle swept counterclockwise from the vertical axis to the object's boundaries, with the sweeping trajectory directed from the outside to the inside of the boundary. These angles range from 0 to 360 degrees as the two sides of the boundary represent different situations, respectively, i.e., covered or uncovered cases. The relationship between the angle of the object and the angle of the edge is expressed by

$$\alpha_p = \mod\left(\alpha + \frac{(p-1)\pi}{2}, 2\pi\right) \quad p = 1, 2, 3, 4 \tag{2.8}$$
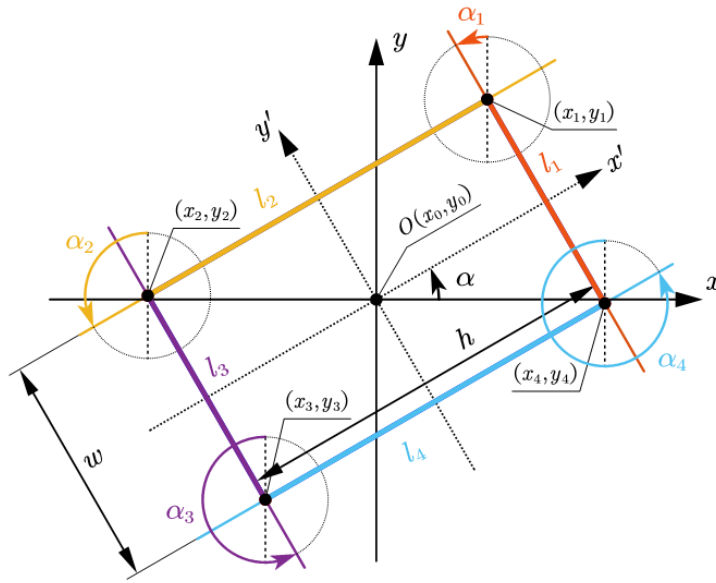
Figure 2.4: Parameters for measured object

The four positions of the object's vertices are numbered according to the quadrant they are located in within the local coordinate system $x'Oy'$. The object's edges in the length direction denoted as l_2 and l_4, are at a distance of $w/2$ from the axis x'. Based on these geometric relations, the corresponding midpoint positions can be derived and applied for edge functions.

$$(x_{12}, y_{12}) = \left(x_0 - \frac{w\sin\alpha}{2}, y_0 + \frac{w\cos\alpha}{2}\right) \tag{2.9}$$

$$(x_{34}, y_{34}) = \left(x_0 + \frac{w\sin\alpha}{2}, y_0 - \frac{w\cos\alpha}{2}\right) \tag{2.10}$$

$$l_2 : \cos\alpha(y - y_{12}) = \sin\alpha(x - x_{12}) \tag{2.11}$$

$$l_4 : \cos\alpha(y - y_{34}) = \sin\alpha(x - x_{34}) \tag{2.12}$$

By combining the above equations, the implicit form of edge functions are written as

$$l_2 : \cos\alpha(y - y_0) - \sin\alpha(x - x_0) - \frac{w}{2} = 0 \tag{2.13}$$

$$l_4 : \cos\alpha(y - y_0) - \sin\alpha(x - x_0) + \frac{w}{2} = 0 \tag{2.14}$$

It should be noted that the object center must be bound by the edge functions. This condition is used to determine the inequality signs in the expression set of S_O from the implicit equations, where $l_2(x_0, y_0) \leq 0$, $l_4(x_0, y_0) \geq 0$. After repeating the above process for the other pair of edges, the region S_O can be finally described as

$$
\begin{aligned}
S_O = \{(x, y) \in \mathbb{R}^2 \mid l_1(x, y) &= \sin\alpha(y - y_0) + \cos\alpha(x - x_0) - \frac{h}{2} \leq 0 \\
l_2(x, y) &= \cos\alpha(y - y_0) - \sin\alpha(x - x_0) - \frac{w}{2} \leq 0 \\
l_3(x, y) &= -\sin\alpha(y - y_0) - \cos\alpha(x - x_0) - \frac{h}{2} \leq 0 \\
l_4(x, y) &= -\cos\alpha(y - y_0) + \sin\alpha(x - x_0) - \frac{w}{2} \leq 0\}
\end{aligned}
\tag{2.15}
$$

For the receiving fiber positioned at (x_R, y_R), The effective light reflection zone S where light can be observed is the intersection between the detection scope $S_R = \{(x, y) \in \mathbb{R}^2 \mid (x - x_R)^2 + (y - y_R)^2 \leq r_0^2\}$ and the region of reflective surface S_O. The plane region S is highlighted in blue in Figure 2.5, where the detection scope is split by a specific edge in the reflective region.
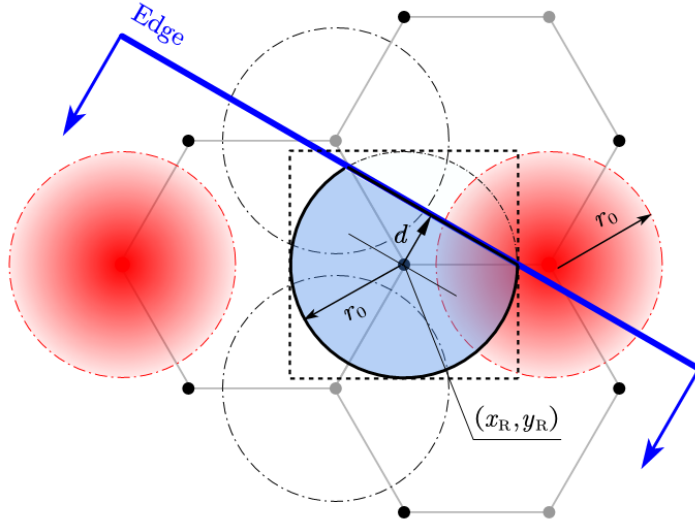

Figure 2.5: Effective light reflection zone for the receiving fiber

The light profile within the effective detection scope is influenced not only by the distance from the receiving fiber to the edge d and the absolute angle α_p, but also by which group of transmitting fibers is illuminated. Thus, a new variable θ is introduced to represent the relative angle between the edge and the illuminated transmitting fiber. This variable represents the degree of freedom for measuring the object's local position in the rotational direction within the plane when considering only a single set of fiber transceivers.

Figure 2.6 shows the relationship between the relative angle θ and the absolute angle α_p of the edge under different illumination cases. Taking the receiving fiber as the origin, the direction toward the illuminated transmitting fiber is defined as the positive horizontal direction x'' of the local coordinate system for the sensing plane. The relative angle θ swept counterclockwise from the vertical axis y'' to the edge directed from outside to inside of the edge. In the first illumination case where the local coordinate coincides with the global coordinate, θ is equivalent to α_p. When the axis y'' is parallel to the edge illustrated as the specific case in Figure 2.6c, the relative angle returns zero.

$$
\theta(\alpha, p, t) = \begin{cases} \alpha_p & t = 1 \\[2mm] \alpha_p + \dfrac{4\pi}{3} - 2\pi\epsilon(\alpha_p - \dfrac{2\pi}{3}) & t = 2 \\[2mm] \alpha_p + \dfrac{2\pi}{3} - 2\pi\epsilon(\alpha_p - \dfrac{4\pi}{3}) & t = 3 \end{cases}
\tag{2.16}
$$

$$\epsilon(x - x_0) = \begin{cases} 0 & x < x_0 \\ 1 & x \geq x_0 \end{cases} \tag{2.17}$$

The final expression of the relative angle is stated in Equation 2.16, where the delayed unit step function $\epsilon(x)$ is applied to limit the range of θ in $[0, 2\pi)$. This definition will be used to decode the fiber outputs as it returns a unique light profile in the detection scope.

(a) $t = 1$

(b) $t = 2$

(c) $t = 3$
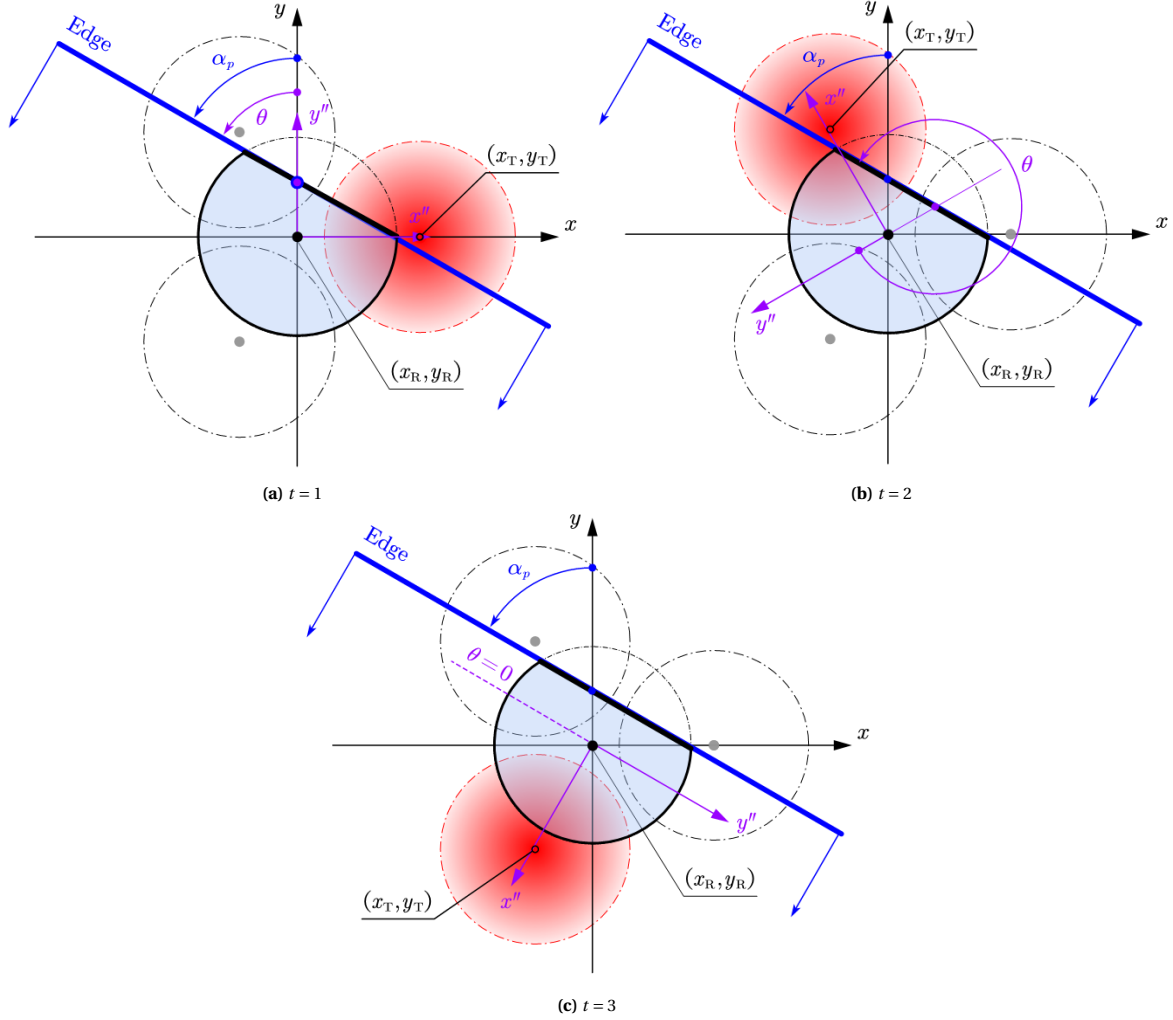
Figure 2.6: Edge orientation θ relative to the illuminated transmitting fiber

2.2.3. Light power output

The simulated light power P can be calculated by a surface integral of light intensity over the effective light reflection zone. The boundaries of the solid arc are transformed into rectangular bounds illustrated as the dashed block in Figure 2.5 by a delta function, where the intensity of light outside the plane region S is set to 0.

$$P(x_0, y_0, x_R, y_R, \theta) = \iint_S I(x, y) \, dx \, dy = \int_{x_R - r_0}^{x_R + r_0} \int_{y_R - r_0}^{y_R + r_0} I(x, y) \delta(x, y) \, dx \, dy \tag{2.18}$$

$$\delta(x, y) = \begin{cases} 1 & (x, y) \in S \\ 0 & (x, y) \notin S \end{cases} \tag{2.19}$$

From Equation 2.18, the overlapping region $S = S_O \cap S_R$ directly determines the power output from the receiving fibers. The maximum light power denoted P_U, is observed when the detection scope S_R is contained

within the reflective region S_O, while no intersection between the above two sets returns the minimum light power $P_L = 0$. The only unknown parameter C_0 in $I(x, y)$ is eliminated by normalizing the light power, see Equation 2.20.

$$N = N_E(x_0, y_0, x_R, y_R, \theta) = \frac{P(x_0, y_0, x_R, y_R, \theta)}{P_U} \in [0, 1] \tag{2.20}$$

The empirical expression of normalized light power N_E can be numerically solved by MATLAB, while it contains a series of arguments. To simplify the expression of the relative position between the object and the sensor unit, the coordinates of the object center and receiving fiber are substituted by the distance between the edge and receiving fiber d. This argument is also normalized by the diffusive radius r_0. The negative terms in the length ratio d/r_0 indicate that the receiving fiber is bounded in the reflective region by the edge.

The numerical results of normalized power in different edge orientations θ is illustrated in Figure 2.7. They have many promising features in position decoding: the light power is monotonously decreased within the detection scope in a smooth pattern, which makes it possible to find the corresponding inverse function. The pairs of curves in a central symmetric pattern can be found, with the relation expressed in Equation 2.21. The Gaussian light intensity distribution is also symmetric along the line connecting the positions of fiber transceivers, with derived expression in Equation 2.22. These relations can be used to check the feasibility of the proposed curve-fitting functions. Though one measured light power correlates to a range of edge positions, the position set of receiving fibers in the sensor array with transitional relative light power indicates the orientation, and this approach will be introduced in the subsequent chapters.

$$N_E\left(\frac{d}{r_0}, \theta\right) = 1 - N_E\left(-\frac{d}{r_0}, \pi + \theta\right) \tag{2.21}$$

$$N_E\left(\frac{d}{r_0}, \theta\right) = N_E\left(\frac{d}{r_0}, 2\pi - \theta\right) \tag{2.22}$$
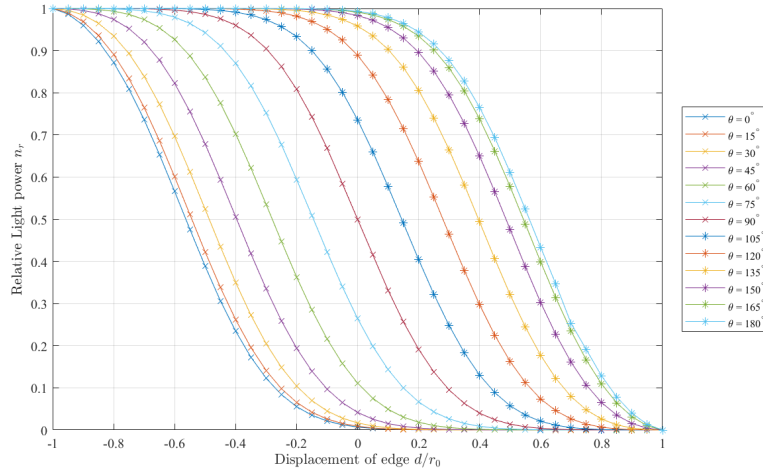
Figure 2.7: Theoretical fiber output N_E in different orientations

2.3. Fiber output fitting

The theoretical power output spends great computation load on solving non-elementary integral functions in Equation 2.18. The approximate mathematical relation in the form of elementary functions between the planar DoFs of the measured object, sensor surface parameters, and normalized light power N will be determined to replace complicated integrals.

The simplest form of multivariate function N_F can be expressed as

$$N = N_F\left(\frac{d}{r_0}, \theta, \frac{r_c}{r_0}\right) \tag{2.23}$$

The inputs are transformed into dimensionless variables to match the normalized output N where the fiber probe density r_c and edge distance d are referred to the diffusive radius r_0. A nested hierarchy is assumed to

separate independent variables. In this case, the multivariate function can be rewritten as $N_F = F\left(\dfrac{d}{r_0}\right)$, where the coefficients of $F\left(\dfrac{d}{r_0}\right)$ are functions of $G(\theta)$, and the coefficients of $G(\theta)$ are functions of $H\left(\dfrac{r_c}{r_0}\right)$.

2.3.1. Edge positions

For the first function layer F, S-shaped curves between two constant levels can be observed in theoretical outputs and these features are satisfied by the Sigmoid function in the basic form of

$$F(x) = \frac{1}{1 + \exp(x)} \tag{2.24}$$

The basic form of the Sigmoid function is asymptotic to 1 for negative infinite input and 0 for positive infinite input. The function can also be altered by adding additional parameters G_1 and G_2 in Equation 2.25, where G_1 changes the slope of transition and G_2 shifts the function.

$$F(x) = \frac{1}{1 + \exp\left(G_1(x - G_2)\right)} \tag{2.25}$$

The Curve Fitting Toolbox in MATLAB is applied to figure out the optimal coefficients G_1 and G_2 with the given fit type $F(x)$ that returns minimum root mean square error for different edge orientation θ. The fitted models are illustrated as dash-dotted lines in Figure 2.8. The proposed function model has a very good fitness to the theoretical results in cross or star markers, with the lowest coefficient of determination (R^2) being no less than 0.99. However, an observable deviation of fitting curves from analytical solutions appears when the relative light power approaches the maximum or minimum value. In these ranges, a tiny difference in light power corresponds to a wide range of the edge position, which adversely affects the stability of measurement. According to the fitting results, the useful range of sensor outputs is confined by

$$0.05 \leq N \leq 0.95 \tag{2.26}$$
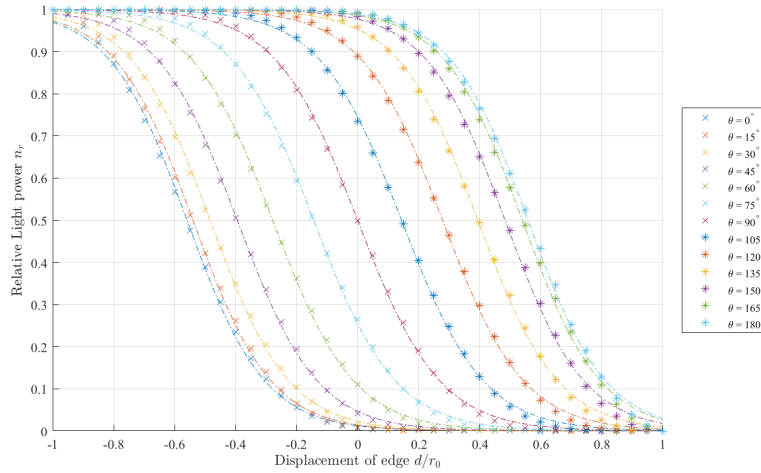
Figure 2.8: Fitting results for $N_F = F\left(\dfrac{d}{r_0}\right)$ in different orientations ($\dfrac{r_c}{r_0} = 1$)

2.3.2. Edge orientations

The optimized coefficients G_1 and G_2 are the functions in the second layer that are dependent on edge orientation θ. These coefficients are plotted in Figure 2.9 as scattered markers for several possible combinations of fiber probe density r_c and diffusive radius r_0. The trigonometric fit type outperforms other elementary function models, where only three coefficients are required to define all functions in high precision. These functions are expressed as

$$G_1(\theta) = H_1 \cos(2\theta) + H_2 \tag{2.27}$$

$$G_2(\theta) = H_3 \cos(\theta) \tag{2.28}$$

From the fitting result of $G_1(\theta)$, the fitting deviation appears when the radius ratio determined by fiber probe density and diffusivity exceeds 1.5 and continues to increase. This could be attributed to a smaller overlapping region between the detection scope and light profiles for a longer distance between the fiber transceivers. To ensure sufficient light profile collected by receiving fibers with good fitness, the radius ratio is limited by

$$\frac{r_c}{r_0} \le 1.5 \tag{2.29}$$

The proposed function models are substituted into the Equations 2.21 and 2.22 to verify their feasibility. Based on the geometric properties of trigonometric functions, G_1 and G_2 should satisfy:

$$G_1(\theta) = G_1(2\pi - \theta) = G_1(\pi + \theta) \tag{2.30}$$
$$G_2(\theta) = G_2(2\pi - \theta) = -G_2(\pi + \theta) \tag{2.31}$$

In this case, Equation 2.22 is verified as all the terms related to θ in the expression of N_F satisfy this condition. By combining Equations 2.25, 2.30 and 2.31, Equation 2.21 can also be proved with the following derivation.

$$N_F\left(-\frac{d}{r_0}, \pi+\theta\right) = \frac{1}{1+e^{G_1(-\frac{d}{r_0}+G_2)}} = \frac{e^{G_1(\frac{d}{r_0}-G_2)}}{e^{G_1(\frac{d}{r_0}-G_2)}+1} = 1 - \frac{1}{e^{G_1(\frac{d}{r_0}-G_2)}+1} = 1 - N_F\left(\frac{d}{r_0}, \theta\right) \tag{2.32}$$
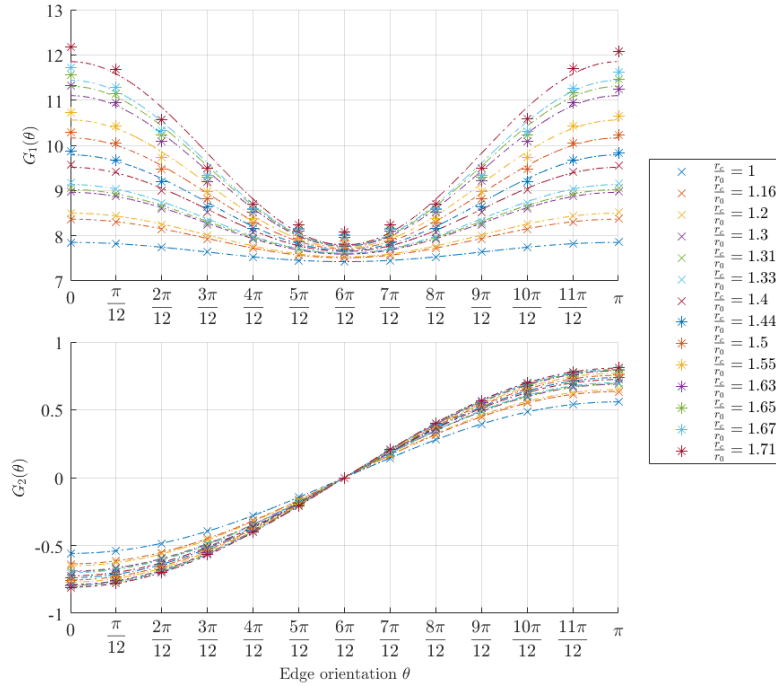
Figure 2.9: Fitting results for $G_1(\theta)$ and $G_2(\theta)$

2.3.3. Sensor surface parameters

The coefficients of trigonometric fitting curves in Figure 2.9 for different sensor parameters are scattered in Figure 2.10. Quadratic polynomials are adopted as fitted models and they accurately reflect the trends of the sample data in cross markers. These polynomials are denoted as **H** in the third function layer of N_F.

$$\mathbf{H}\left(\frac{r_c}{r_0}\right) = \begin{bmatrix} H_1 \\ H_2 \\ H_3 \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} (\frac{r_c}{r_0})^2 \\ \frac{r_c}{r_0} \\ 1 \end{bmatrix} \tag{2.33}$$

The numerical results of the coefficient matrix in Equation 2.34 are returned with 95% confidence bounds. Four significant digits are retained and the resultant R^2 is close to 1 in the goodness of the fit.

$$\mathbf{c} = \begin{bmatrix} 2.367 & -3.894 & 1.744 \\ 2.233 & -2.989 & 8.391 \\ 0.2032 & -0.8997 & 0.1378 \end{bmatrix} \tag{2.34}$$

Noted that **H** is a constant in the measurement setup when the optimized fiber probe density and diffusivity are determined while the planar DoFs d and θ are varied by the position of the measured object. If the orientation is decoded by the fiber optic sensor array, the predicted distance $\widehat{d}$ between the receiving fiber and the edge of the object can be solved by the inverse function of the fitting model N_F.

$$\widehat{d} = N_F^{-1}(N, \theta) = r_0 \left(\frac{1}{H_1 \cos(2\theta) + H_2} \ln\left(\frac{1 - N}{N} \right) + H_3 \cos\theta \right) \tag{2.35}$$
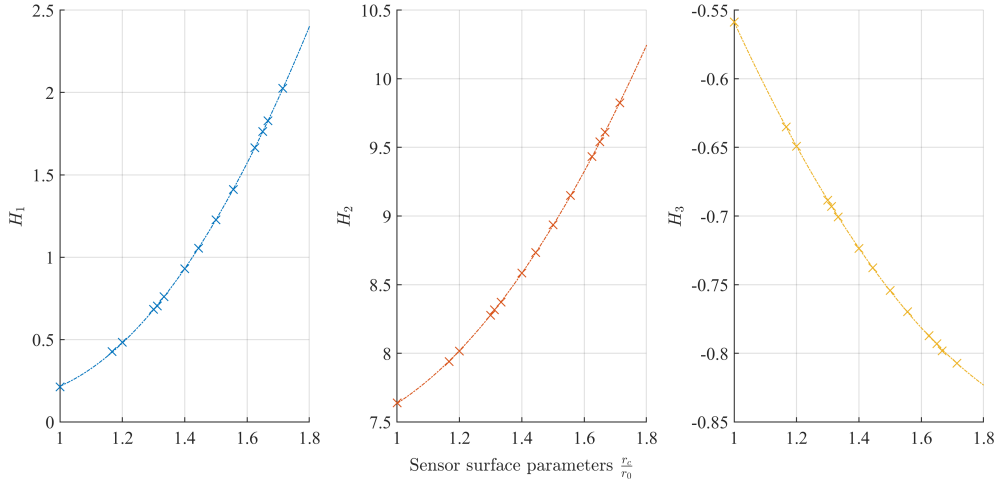
Figure 2.10: Fitting results for $H_1\left(\dfrac{r_c}{r_0}\right)$, $H_2\left(\dfrac{r_c}{r_0}\right)$ and $H_3\left(\dfrac{r_c}{r_0}\right)$

2.4. Diffusive radius evaluation

The remaining undetermined parameter r_0 in Equation 2.35, will be derived based on critical positions in the measured sensor output curve that are asymptotic to maximum or minimum light power N. The layout for the object edge with blue arrows pointed inward the boundary, the light spot and detection scope centered at (x_T, y_T) and (x_R, y_R) respectively are plotted in Figure 2.11 for clear description.
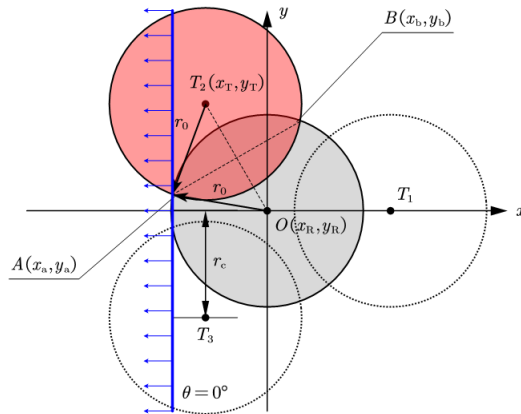
Figure 2.11: Geometric relations between diffusive radius r_0 and critical positions x_a, x_b

In a Cartesian coordinate system with the center of the receiving fiber as the origin, the radius of the light spot, which is equivalent to the radius of the detection scope according to the definition of fiber diffusivity in Section 2.1, can be expressed as:

$$r_0^2 = x_a^2 + y_a^2 = (x_a - x_T)^2 + (y_a - y_T)^2 \tag{2.36}$$

This involves an unknown parameter y_a. As the line segments AB and OT_2 are mutually perpendicular bisectors of each other with given coordinates of fiber transceivers, the product of the slopes of the two segments is -1 and the midpoint $(\frac{x_T}{2}, \frac{y_T}{2})$ is used to construct the line equation of AB.

$$l_{OT_2} : y = \frac{y_T}{x_T} x \tag{2.37}$$

$$l_{AB} : y - \frac{y_T}{2} = -\frac{x_T}{y_T}\left(x - \frac{x_T}{2}\right) \tag{2.38}$$

The above expression l_{AB} returns y_a in terms of x_a, x_T and y_T. Thus, when the critical position x_1 is determined asymptotically with the maximum light power, the diffusive radius can be solved by

$$r_0 = \sqrt{x_a^2 + \left(\frac{y_T}{2} - \frac{x_T}{y_T}\left(x_a - \frac{x_T}{2}\right)\right)^2} \tag{2.39}$$

As the midpoint $(\frac{x_T}{2}, \frac{y_T}{2})$ also bisects line segment AB, an extra constraint between critical positions and coordinate of transmitting fiber in the horizontal axis is added, which could help find the transitional range $[x_a, x_b]$ more accurately.

$$x_a + x_b = x_T \tag{2.40}$$

2.5. Conclusion

Based on existing research results, this chapter adopts a hexagonal grid arrangement of optical fibers, considering fiber probe density and diffusivity as the main design variables for the sensing platform. A simulated fiber array is established, and the theoretical fiber output is derived by performing a surface integration of Gaussian-distributed light within the effective light reflection zone. The results indicate that, due to the light power output corresponding to multiple measurement results with different object edge angles and positions, it is impossible to identify the object's position with only one set of transmitting and receiving fibers. However, if the object's angle can be determined in advance, its boundary position can also be ascertained. The fiber array can serve as a breakthrough for this limitation. In this array, the positions of the receiving fibers with light-power output can be used to estimate the object's angle, thereby solving for the object's planar degrees of freedom. Through numerical fitting, an expression is obtained for the edge position of the measured object about the sensor results and the design parameters of the sensing platform. Some of the undetermined design parameters, such as the diffusivity radius, are derived from the characteristics of the calibrated light power-displacement curves.

<div align="right">

3

</div>

Sensor output calibration

In this chapter, we validate the concept of position measurement using fiber optic sensors through a prototype experiment involving a single set of fiber transceivers. Instead of relying on theoretical light profiles, we measure the actual amount of light captured by the receiving fiber, which is generated by red LEDs, using a CCD camera. The corresponding displacement of the object in planar DoFs is also calibrated to verify the relationships derived from theoretical formulas. The optimal sensor configuration identified through these experiments will be utilized in the subsequent chapters.

3.1. Experimental devices

3.1.1. Optical fiber

In this research, plastic optical fiber is utilized, as illustrated in Figure 3.1. The fiber core is composed of polymer PMMA and is coated with a thin layer of fluorine polymer, which has a lower refractive index than the core. The fiber, comprising both the core and cladding, has a diameter of 1 mm and is encased in a polyethylene jacket with an outer diameter of 2.2 mm.
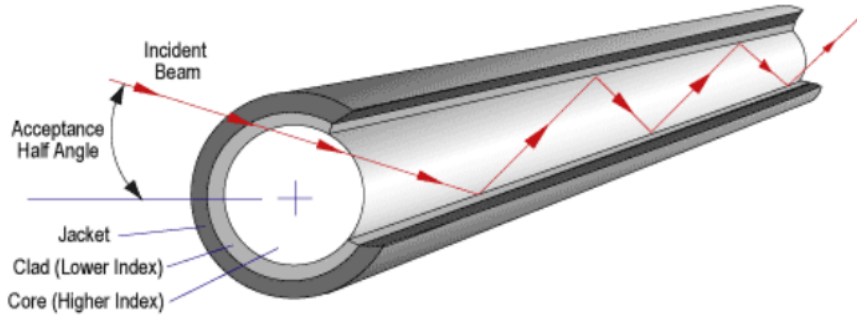
Figure 3.1: Optical fiber[39]

Optical fibers are cut into segments using blades. However, due to blade wear and other factors, the cut surface of the fiber core may sometimes have cracks, which can affect the actual light profiles. To minimize these cracks, fine sandpaper is used to polish the fiber surfaces when only a few fibers are needed for the experiment. In the fabrication of the sensor surface, where hundreds of fibers may be required, a milling machine is employed for processing fiber surfaces.

3.1.2. Motion control mechanism

The actual position of the reflective object is the prime issue for calibration. This problem was addressed in previous studies and was solved by moving a slider in steps with protruding trapezoids[4] or by customizing reflective surfaces in a series of dimensions[36]. The former case has better repeatability due to well-defined

object positions, while the latter allows the object placement in multiple directions. Both merits will be considered to further improve the motion control mechanism in this research. The expectations are as follows:

- Relative in-plane motion between object and sensor unit
- Sealing light from the environment
- Easy for adjusting and measuring real object positions
- Limited assembly tolerance

To meet these requirements, the setup was firstly designed as illustrated in Figure 3.2. The mirror surface tailored from reflective foils is adhered to the bottom surface of the slider. The transparent slider is matched to the grooves in the green basement, which allows continuous linear motion. The protruding rectangular blocks in the front surface of the basement and slider are used to localize the fixed jaw and sliding jaw of the vernier caliper respectively so that the actual linear displacement can be measured in micron scale resolution. The slider and the basement also form an isolated chamber above the sensor surface in which the light only emits from three transmitting fibers in the surrounding of the receiving fiber. The orange fiber transceiver holder can be placed in discrete and well-defined orientations, with the out-of-plane displacement in the axial direction fixed by an extra elastic O-ring in the groove of the fiber holder and the radial motion constrained by the rectangular splines, see the perspective view from the front and top in Figure 3.2b and 3.2c.

(a) Isometric view

(b) Front view

(c) top view
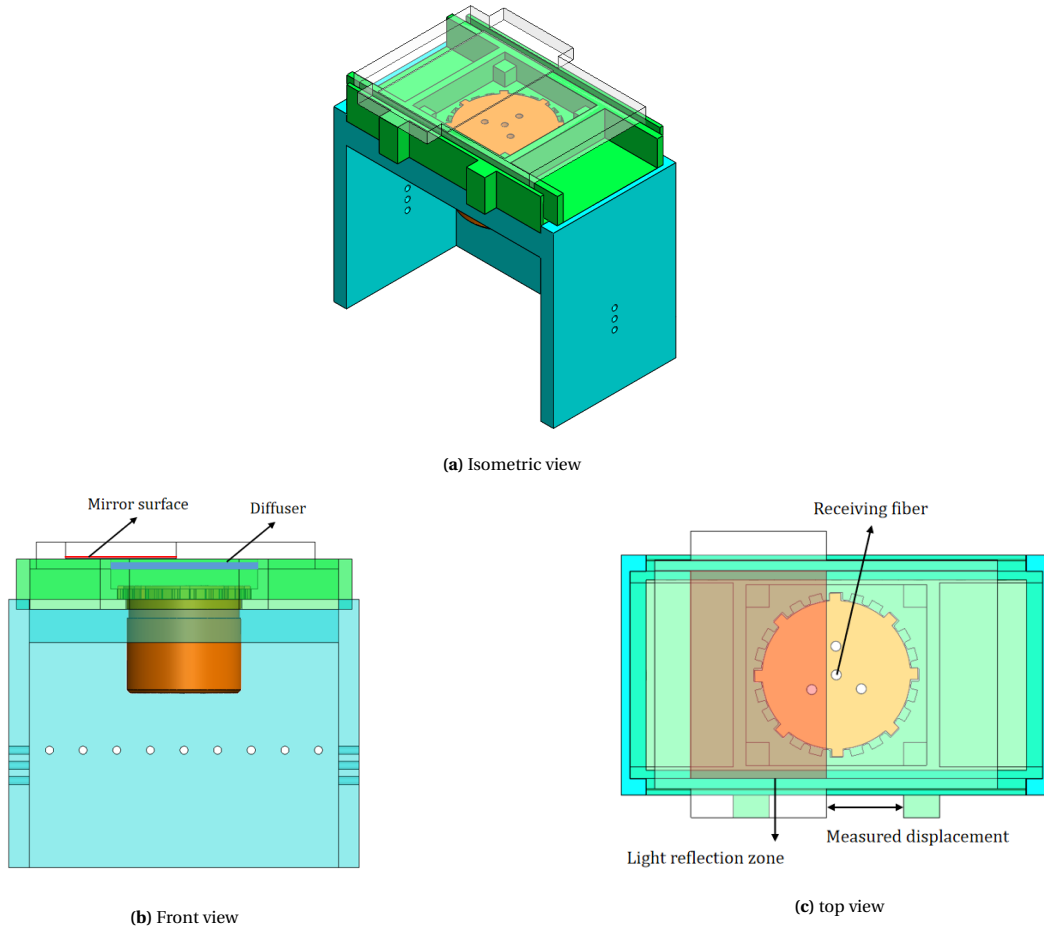
Figure 3.2: SOLIDWORKS models for experimental setup

In addition to motion control, sufficient space is left to accommodate the necessary optical components. The holes on the cyan supporting frame of the mechanism are used to fix the transmitting fibers with an adjustable bending radius and this could also effectively reduce the bending stress exerted on the fiber holder. The fiber holder in this mechanism can be easily replaced by counterparts with distinct distances between

fiber transceivers to explore how real sensor performance depends on fiber density r_c. The configuration of the mechanism also attempts to improve the diffusivity r_0 by introducing air gaps between the sensor surface, diffuser, and mirror surface at the millimeter level.

3.1.3. Illumination control module

To illuminate three groups of transmitting fibers in a time sequence, two prime factors are considered in the electric circuit design and these will be explained in this subsection.

Firstly, the luminous intensity of light sources related to different transmitting fibers should be adjustable independently to guarantee the consistency of the light spots. This can be achieved by the common emitter amplifier circuit illustrated in Figure 3.3, where the current flowing through the light source I_c is controlled by the base current I_b. However, the amplifier circuit functions only properly when the transistor is in the active state, and it depends greatly on the value of the resistors R_p and R_b, which will be determined later on.
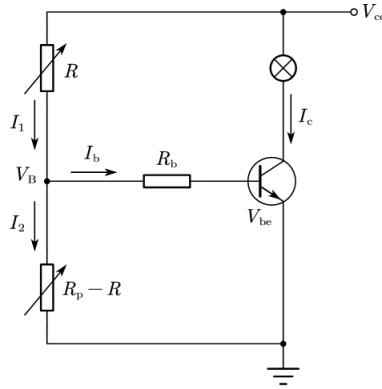
Figure 3.3: Common-emitter amplifier circuit for illumination control module

The base of the transistor is connected to the potentiometer wiper with a total resistance R_p so that the branch current I_b changes with the position of the wiper. The mathematical relations between the annotated variables are expressed in following equations.

$$I_1 = \frac{V_{cc} - V_B}{R} \tag{3.1}$$

$$I_2 = \frac{V_B}{R_p - R} \tag{3.2}$$

$$I_b = \frac{V_B - V_{be}}{R_b} \tag{3.3}$$

$$I_1 = I_2 + I_b \tag{3.4}$$

$$I_c = \beta I_b \tag{3.5}$$

By eliminating the intermediate variables, the base current can be expressed as:

$$I_b = \frac{V_{cc}(R_p - R) - V_{be}R_p}{R_b R_p + R(R_p - R)} \quad R \in [0, R_p] \tag{3.6}$$

For non-positive base current, the transistor works in cut-off state and there is no current in the collector branch. In this case, the relative position of the wiper in the potentiometer is limited by

$$I_b > 0 \Rightarrow \frac{R}{R_p} < 1 - \frac{V_{be}}{V_{cc}} \tag{3.7}$$

V_{be} is the threshold voltage of the conducting base-emitter junction in the transistor, which is a constant with approximately 0.6 V for silicon diodes[47].
The collector current I_c saturates and does not depend on I_b anymore when the amplified base current βI_b

exceeds the maximum allowable current $I_{c,max}$. To ensure that I_c is adjustable in a certain range, the base current is limited at the wiper's mirror position relative to the transistor's cutoff state. This constraint corresponds to Equation 3.8 and the relation between R_b and R_p is derived in Equation 3.9.

$$I_b\left(R = \frac{V_{be}}{V_{cc}}R_p\right) \le \frac{I_{c,max}}{\beta} \tag{3.8}$$

$$R_b \ge \beta\frac{V_{cc} - 2V_{be}}{I_{c,max}} - \frac{V_{be}(V_{cc} - V_{be})}{V_{cc}^2}R_p \tag{3.9}$$

The red LED lamp[44] and the transistor[48] are applied in the circuit. According to the provided data sheet, the maximum forward current in LED $I_{c,max}$ = 50 mA and the nominated current $I_{c,nom}$ = 20 mA. the minimum DC gain for the transistor β = 60. These electronic elements are power supplied by digital output pins of micro-controller unit Arduino, with nominated voltage V_{cc} = 5 V. When the potentiometer with a total resistance R_p = 50 kΩ is selected, the minimum allowable resistance for resistor $R_{b,min}$ = 72 Ω.

To ensure that the adjustable range of the operating current of the light source is near the rated current $I_{c,nom}$, the lower limit of the base current at the critical position is also constrained. The relation derived in Equation 3.11 results in the maximum allowable resistance $R_{b,max}$ = 420 Ω. Thus, one of the standard base resistor values R_b = 160 Ω in the feasible range is chosen to construct the illumination module.

$$I_b\left(R = \frac{V_{be}}{V_{cc}}R_p\right) \ge \frac{2I_{c,nom}}{\beta} \tag{3.10}$$

$$R_b \le \beta\frac{V_{cc} - 2V_{be}}{2I_{c,nom}} - \frac{V_{be}(V_{cc} - V_{be})}{V_{cc}^2}R_p \tag{3.11}$$

Based on the amplifier circuit and determined electronic components, the illumination control module for all channels of light sources in a sensor unit is constructed, see the connection diagram in Figure 3.4. The Arduino board is connected to the computer through the USB jack so that the power supplies in separate output channels ~9 - ~11 can be switched on/off through programs executed by the computer.
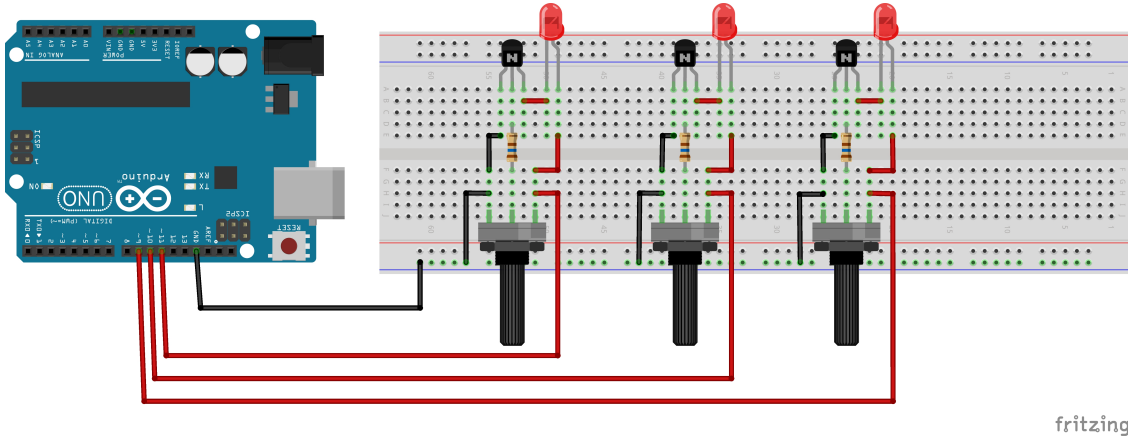
Figure 3.4: Circuit connection diagram for electronic elements in illumination control module

3.1.4. Setup realization

The measurement setup for a single set of fiber transceivers in the sensor array is built up and illustrated in Figure 3.6. This measurement device aims to verify the proposed light propagation model and determine the main design parameters in the sensor array, and the corresponding experiment is carried out in the following section.

The electronic elements in the illumination control module except light source and power supply are welded on PCB with additional mounting screw terminal connectors. In addition to compactness, this integrated circuit can be used to control other light sources with suitable voltage inputs, such as the LED array for distributed fiber optic sensors. The connector also makes it easier to measure the voltage of the light source using a multimeter, so that the rotary shaft of the potentiometer can be adjusted with a resolution of 1 mV.
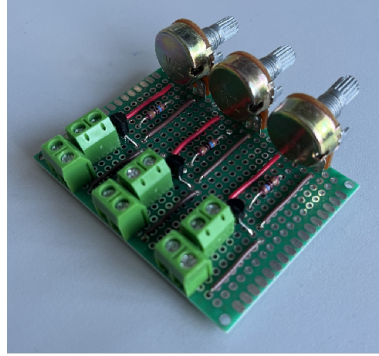
Figure 3.5: Integrated circuit for light illumination control module

The parts in the motion control mechanism are manufactured by the FDM 3D printer, with a minimum tolerance of 0.1 mm. This structure, together with the project board below, is fixed to the ground to avoid disturbance. The slider is black to absorb most of the light out of the reflective region of its bottom surface, as shown in Figure 3.6b. Three LED lamps installed on the project board are aligned to the transmitting fibers by hollow blockers, and the receiving fiber is aligned to the CCD camera by the cube block with a chamber inside to isolate the ambient light.
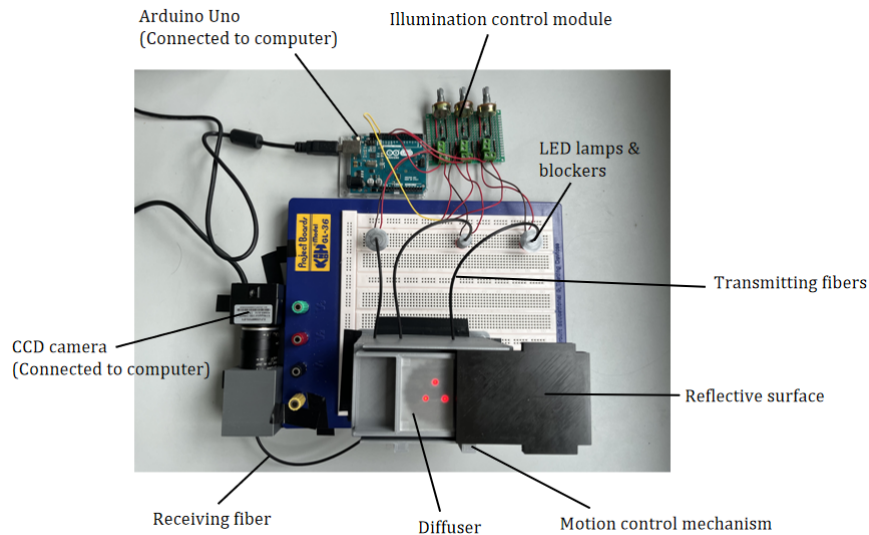
(a) Main elements and their configurations in the realized setup
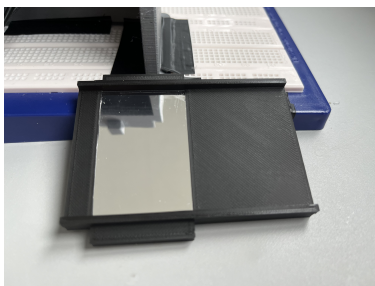
(b) Reflective surface from bottom view
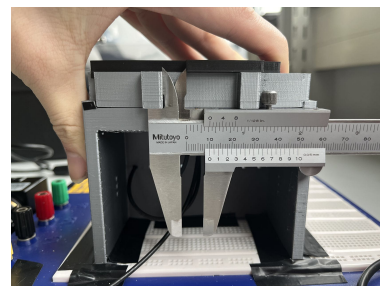
(c) Surface positioning by vernier caliper

Figure 3.6: The realized setup for light power - displacement curve calibration

The sensor unit is entirely covered by a reflective surface before the calibration of the sensor output to the

edge position. In this case, the potentiometers are regulated until the discrepancies in sensor output values for different illuminated transmitting fibers are minimized. During the calibration process, the vernier caliper with 0.05 mm resolution is used to localize the real position of the reflective surface. The output values corresponding to 0.25 mm-step displacements from fully covered to uncovered cases for distinct illuminated transmitting fiber are recorded.

3.2. Experimental procedure

3.2.1. Image acquisition and processing

The type of camera applied in this research is USB webcam ELP-USB8MP02G-SFV. It has a varifocal lens from 2.8 mm to 12 mm, allowing a compact design to integrate the camera into the measurement setup. This USB webcam is compatible with the image processing applications in MATLAB so that the snapshots are taken automatically and useful information can be extracted from the images through the program.

The colored image captured by this camera consists of an array of 1600 × 1200 pixels, with each pixel containing intensity values ranging from 0 to 255 in the red, green, and blue channels. The camera's manual exposure and white-balance settings are configured to prevent automatic adjustment of the light projected onto the photosensitive layer when the intensity exceeds the maximum RGB value. All circular light spots within the specified radius range in the image can be detected using a MATLAB function **imfindcircles**(image, radius range), which returns the center position (u_R, v_R) and radius r_R for each detected light spot.
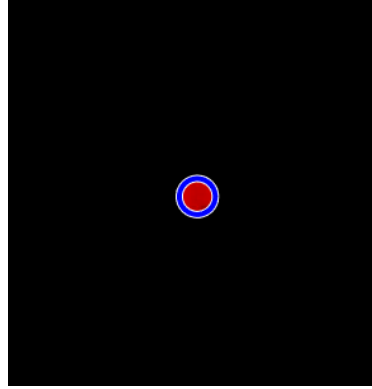
Figure 3.7: Light pattern of a receiving fiber shot by CCD camera, with identified spot center and radius (blue circle)

As shown by the blue circle in the partial view of the snapshot in Figure 3.7, the function achieves sufficient accuracy in detecting light spots and the average RGB value of the pixels constrained by the identified circle will be used as the sensor output. In reality, the measured results in the G and B channels are negligible compared to the counterparts in the R channel when the red LED lamps are used as the light source. Overall, the real light power $\widehat{P}$ detected by a receiving fiber in an image can be evaluated by the following expression:

$$\widehat{P} = \widehat{P}(\text{image}, u_R, v_R, r_R) = \frac{\sum\limits_{(u,v)\in \hat{S}_R} R(u,v)}{|\hat{S}_R|}, \quad \hat{S}_R = \left\{ (u,v)|(u - u_R)^2 + (v - v_R)^2 \leq \frac{1}{2} r_R^2 \right\} \tag{3.12}$$

where $R(u,v)$ represents the R value of the pixel located at column u and row v in the given image and $\hat{S}_R$ is the set of effective pixels to be averaged. This range is smaller than the enclosed region by the identified circle to reduce the impact of variation of the center position on the measurement result.

3.2.2. Diffuser type selection

The main function of the light diffuser in the sensing plane is to spread the emitted light as much as possible with minimal loss. Among light diffusers made from different types of material, acrylic diffusers are extensively used in lighting fixtures, displays, signing, and profiles. Due to its low cost, 100% colorlessness, ease of fabrication, and barely any thickness tolerance[45], this material is chosen as the primary candidate for the light diffusers in this study.

Under the same material, light diffusers are classified into different products based on properties such as

light transmittance, thickness, and surface characteristics. These properties also affect the performance of the calibrated sensor output curves concerning object displacement. For instance, diffusers with lower light transmittance cause the minimum R-value captured by the receiving fiber to increase, because a higher proportion of emitted light is directly reflected from the diffuser surface. In contrast, thick diffusers decrease the maximum measured light power, as part of the light reflected from the mirror surface is absorbed within the diffusive layer. To ensure sensor output sensitivity with a greater difference between the maximum and minimum measured light power, the recommended diffuser types feature over 70% light transmittance and a thickness within 3 mm in the product details. These conditions return a list of available diffuser types in the following table.

Table 3.1: Candidate diffuser types and technical specifications[45]

Sample number	Diffuser type	Material	Thickness(mm)	Light transmittance
1	PyraLed YT275	Acrylic	1.8	77%
2	PyraLed YT290	Acrylic	1.8	90%
3	PRIMO XT N381	Acrylic	2	71%
4	PyraLed Makrolon DX 1141	Polycarbonate	1.5	76%
5	PyraLed Makrolon DX 1141	Polycarbonate	3	76%

On the other hand, these diffusers require a slightly greater distance from the light source to spread the light diffusely. In the measurement setup illustrated in Figure 3.2b, there is a 5 mm air gap from the fiber tips to the bottom surface of the diffuser and a 3 mm distance from the bottom surface of the diffuser to the mirror surface. The gaps between surfaces will be replaced by transparent PMMA plates in the sensor array configuration. To evaluate the diffusivity, the sensor output curves with different types of diffusers are calibrated, as illustrated in Figure 3.8. The output curve without diffusers is also measured as a reference.

(a) Illuminated transmitting fiber T_1, $\theta = 0°$

(b) Illuminated transmitting fiber T_2, $\theta = 120°$
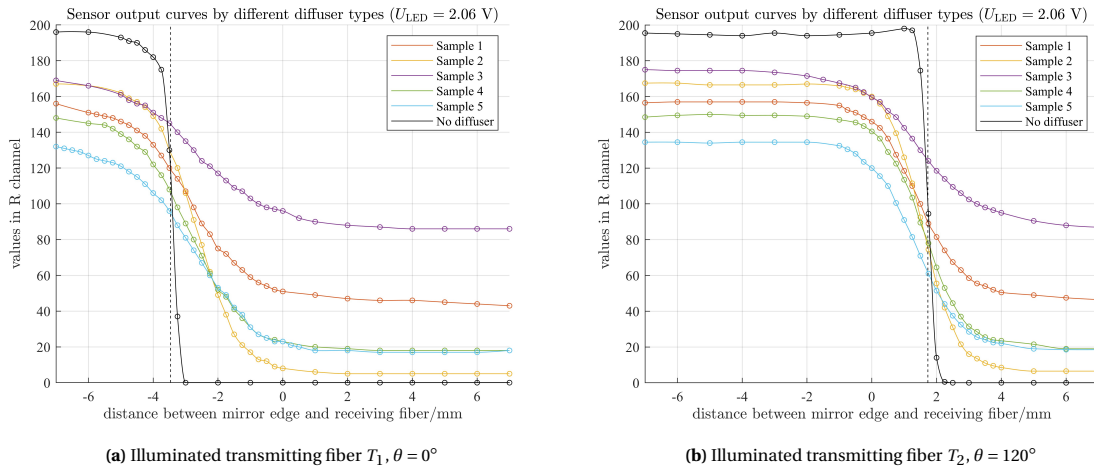
Figure 3.8: Calibrated sensor output curves by different types of diffusers ($r_c = 6$ mm)

From a sensor performance perspective, applying the diffuser results in smoother curves, while there is no significant differences in the diffusivity among various diffuser types. For acrylic diffusers of similar thickness, the most notable variation in sensor output curves from samples 1 to 3 occurs at the minimum R values, where higher light transmittance leads to a lower minimum value. For the same diffuser type with different thicknesses, the curves for samples 4 and 5 nearly overlap at their minimum values, though the thicker sample exhibits a lower maximum value. Overall, sample 2 shows the greatest difference between maximum and minimum values among all output curves, offering higher resolution for position decoding. Therefore, this diffuser type will be used in subsequent experiments for this research.

The set of measured curves also indicates the feasibility of the theoretical light profiles proposed in Chapter 2 to some extent. The smooth S-shaped curves are obtained, with a similar trend of curve shift for different angles θ. The fitness of the obtained curve to theoretical model will be evaluated when the real diffusive

radius r_0 is determined. The global coordinates of illuminated transmitting fiber in x axis for the above two cases are

$$x_T(\theta = 0°) = \frac{2r_c}{\sqrt{3}} = 6.93 \text{ mm} \tag{3.13}$$

$$x_T(\theta = 120°) = -\frac{r_c}{\sqrt{3}} = -3.46 \text{ mm} \tag{3.14}$$

When there is no diffuser placed above the fiber transceiver, the critical positions that approach the extreme R values almost coincide. The average center is illustrated as a dashed vertical line, with

$$x_a(\theta = 0°) \approx x_b(\theta = 0°) = -3.5 \text{ mm} \approx -\frac{1}{2}x_T(\theta = 0°) \tag{3.15}$$

$$x_a(\theta = 120°) \approx x_b(\theta = 120°) = 1.7\text{mm} \approx -\frac{1}{2}x_T(\theta = 120°) \tag{3.16}$$

It should be noted that the horizontal axis in the above graphs represents the relative position of the mirror edge to the receiving fiber, where the negative values indicate that the fiber is covered by the object. This defined direction is exactly opposite to the direction of the global coordinate illustrated in Figure 2.11 when the reflective object moves in horizontal direction. According to these results, the relationship between critical positions derived by theoretical model in Equation 2.40 has been proved.

3.2.3. Sensor surface parameters identification

The sensor output curves obtained by different fiber densities r_c are also evaluated by the experiment, where the difference in transitional range of sensor outputs can be observed in Figure 3.9. Following the Equation 2.39, when the measured critical positions with the minimum fiber output are $x_a(r_c = 6 \text{ mm}) = 4 \text{ mm}$ and $x_a(r_c = 8 \text{ mm}) = 7 \text{ mm}$, the corresponding diffusive radii are $r_0(r_c = 6 \text{ mm}) \approx 4.3 \text{ mm}$ and $r_0(r_c = 8 \text{ mm}) \approx 7.2 \text{ mm}$ respectively. These parameters, together with the sensor array configuration, determine the empirical sensor output curves expressed by Equation 2.35. The fitness between measurement results and theoretical models is calculated by the coefficient of determination R^2 with following expression:

$$R^2 = 1 - \frac{\sum_i (\hat{y}_i - y_i)^2}{\sum_i (\overline{y_i} - y_i)^2} \tag{3.17}$$

where $\hat{y}_i$, y_i, and $\overline{y_i}$ represent modelled, observed and the mean of observed value in R channel respectively. An R^2 of 1 indicates that the predictions perfectly fit the data. For the observed data sets marked as yellow circles when $r_c = 6 \text{ mm}$ and $r_c = 8 \text{ mm}$, the goodness of fit reaches 99% and 97% respectively. These results verify the feasibility of the theoretical model of light power, which will be used for position decoding in the following chapters.
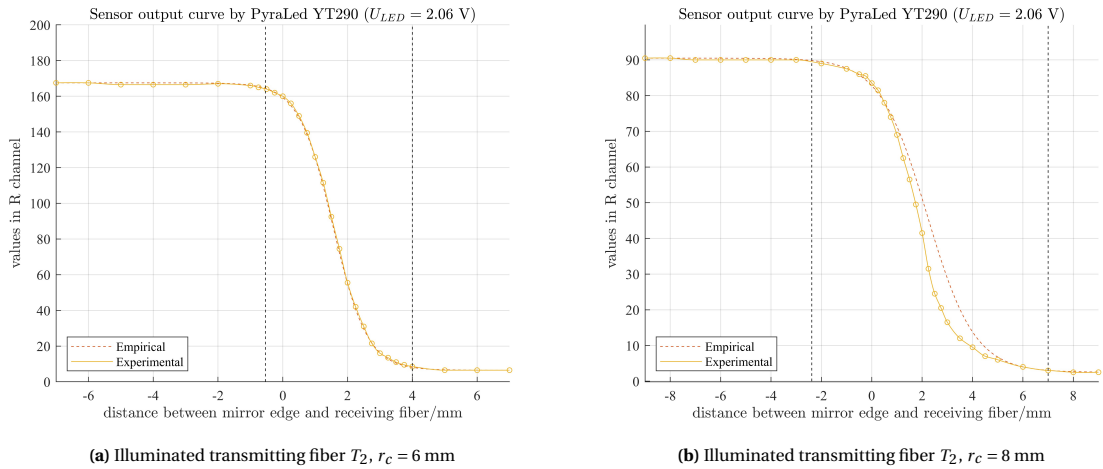

(a) Illuminated transmitting fiber T_2, $r_c = 6$ mm

(b) Illuminated transmitting fiber T_2, $r_c = 8$ mm

Figure 3.9: Calibrated sensor output curves by different fiber densities

The fiber probe density and the resolution were evaluated among fiber holders with different distance between fiber transceivers. As shown in Figure 3.9a, a better fitness can be observed with a greater difference between maximum measured R value $\widehat{P_U}$ and minimum measured R value $\widehat{P_L}$ in sensor output when $r_c = 6$ mm compared to the measured data set in Figure 3.9b. However, this configuration with smaller fiber spacing requires over 200 receiving fibers over an A5-sized measurement plane, nearly twice the number needed when $r_c = 8$ mm.

Table 3.2: Evaluated sensor array properties among different in-radius of hexagon unit r_c

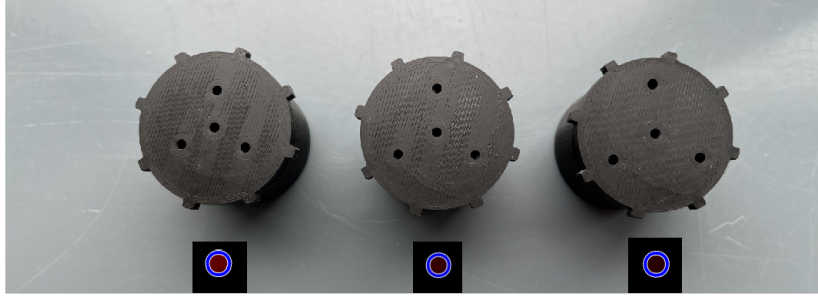| r_c /mm	Number of receiving fibers	Maximum R value $\widehat{P_U}$	Minimum R value $\widehat{P_L}$
6	218	168	7
8	119	91	2
9	91	55	0
10	78	40	0

Figure 3.10: Maximum light power for fiber holder series ($r_c = 8, 9, 10$ mm from left to right)

The achievable resolution is determined by the minimum displacement change per R value. For the above calibrated curves, their resolutions are calculated as:

$$\text{Resolution}(r_c = 6 \text{ mm}) = \frac{|x_b(r_c) - x_a(r_c)|}{\widehat{P_U}(r_c) - \widehat{P_L}(r_c)} = \frac{4 - \left(\frac{6}{\sqrt{at3}} - 4\right)}{168 - 7} \approx 0.028 \text{ mm} \tag{3.18}$$

$$\text{Resolution}(r_c = 8 \text{ mm}) = \frac{|x_b(r_c) - x_a(r_c)|}{\widehat{P_U}(r_c) - \widehat{P_L}(r_c)} = \frac{7 - \left(\frac{8}{\sqrt{3}} - 7\right)}{91 - 2} \approx 0.105 \text{ mm} \tag{3.19}$$

Although the latter is less sensitive than the former, it can still achieve the resolution in sub-millimeter level. As r_c continues to increase, the observed maximum R value decreases to 55 and 40 at $r_c = 9$ mm and 10 mm, respectively. In these cases, the decrease in resolution is significantly greater than the reduction in the number of fibers. Therefore, $r_c = 8$ mm is selected from the dimension series, considering both the accuracy requirements and the cost of the fiber arrangement.

3.3. Conclusion

In this chapter, starting with a single set of transmitting and receiving optical fibers, we calibrated the relationship between object displacement and fiber output using a constructed prototype, thereby validating the theoretical model proposed in Chapter 2. The motion mechanism for calibrating object displacement achieves relative planar movement between the object and the measurement unit through a sliding pair and a spline fit, with a translation resolution of up to 0.05 mm. The illumination of the transmitting fibers is controlled by the digital voltage output pins of an Arduino microcontroller and the corresponding program. The brightness can be manually adjusted using a potentiometer in the common emitter amplifier circuit to ensure consistency in brightness among different light sources. The output curves of the receiving fibers were calibrated under different types of light diffusers and sensor surface parameters. When the inscribed circle radius of the hexagonal unit in the sensor array is 8 mm with the selected diffuser PyraLed YT290, the experimental data achieved a fitting accuracy of 97% compared to the theoretical output, with the resolution around 0.1 mm in a relatively sparse distribution of receiving fibers.

4

Planar positioning algorithms

In the previous chapter, the basic dimensional parameters of the sensor surface have been determined through brightness calibration using a single set of transmitting and receiving fibers. These dimensions are used in this chapter to construct a simulated fiber array sensing platform. The corresponding position recognition algorithm will originate from the simulated platform and be validated on it.

4.1. Algorithm overview

The data flow diagram of the planar positioning algorithm is illustrated in Figure 4.1. In this diagram, all data originates from the external entities enclosed by the rectangular boxes. The rounded rectangles labeled "P" represent the main processing steps, while the open rectangles labeled "D" represent the primary data that are stored and retrieved. The set of fiber positions near the boundary of the object S_{B} and the normalized light power N returned by these fibers serve as inputs for the planar positioning algorithm. These data sets, along with their derivations in the green section, will be replaced in Chapter 6 when the algorithm is applied to the real prototype.
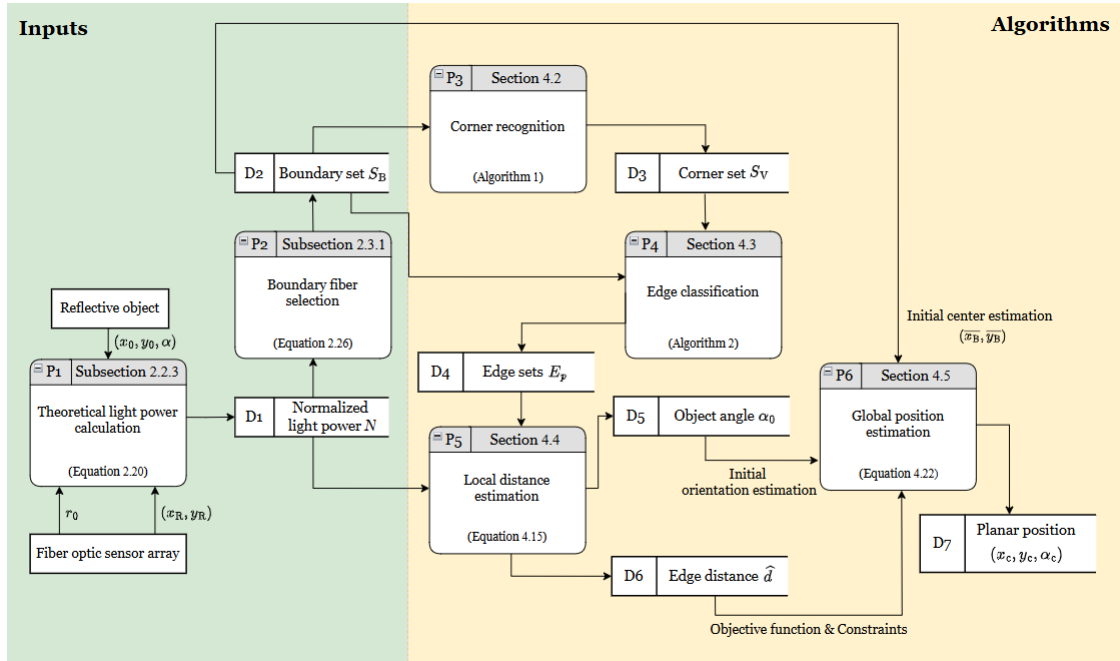

Figure 4.1: Data flow diagram of planar position estimation method by distributed fiber optic sensors

The first process in the section of algorithms, namely the corner recognition, attempts to identify four fiber positions in the boundary set S_B that are nearest to the vertices of the object. This could help divide the rectangular object's boundaries into separate edges, allowing the remaining fibers to be assigned to the nearest edge in edge classification.

In the process of local distance estimation, the inverse function of fitting model N_F^{-1} is applied to figure out the distance $\hat{d}$ from each receiving fiber to the edge. The only undetermined parameter, i.e., the relative angle between the edge and illuminated transmitting fiber, can be estimated by the orientation of fiber positions from the corresponding edge set E_p.

In the final process of the algorithm, the initial solution of the object's planar DoFs at the averaged center of the boundary set $(\overline{x_B}, \overline{y_B})$ and α_0, are updated regarding distance $\hat{d}$ that is derived from the sensor output. In other words, the goal is to find the optimal planar position (x_c, y_c, α_c) such that, under these estimated results, the difference between the geometric distance from each fiber to the object boundary and the distance $\hat{d}$ is minimized. The concept and realization of the above steps will be explained in detail in the following sections.

4.2. Corner recognition

The fiber positions adjacent to object boundaries S_B are used as input in the process of corner recognition. They are selected by the normalized light power with transitional values and are marked by green circles in Figure 4.2.

$$S_B = \left\{ (x_B, y_B) \in \mathbb{R}^2 \mid 0.05 \leq N(x_0, y_0, x_B, y_B, \theta) \leq 0.95 \right\} \tag{4.1}$$
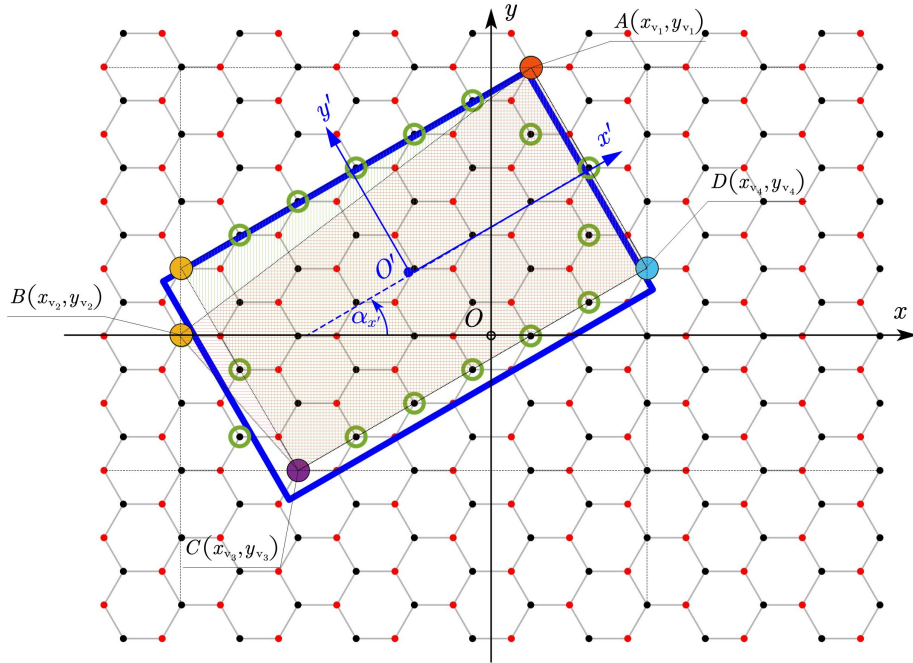

Figure 4.2: Corner recognition process

When the horizontal axis of the sensor surface x is not orthogonal or parallel to that of the object x', fibers near the vertices of the object, namely, the corners, will have an extreme value in either the x or y direction. These extreme values are indicated by colored horizontal or vertical lines. The four corners are allocated to

different subsets, with each subset containing no more than three fibers.

$$
\begin{aligned}
S_{y_U} &= Y_U(S_B) = \{(x, y) \in S_B \mid y = \max(y_B)\} \\
S_{x_L} &= X_L(S_B) = \{(x, y) \in S_B \mid x = \min(x_B)\} \\
S_{y_L} &= Y_L(S_B) = \{(x, y) \in S_B \mid y = \min(y_B)\} \\
S_{x_U} &= X_U(S_B) = \{(x, y) \in S_B \mid x = \max(x_B)\}
\end{aligned}
\tag{4.2}
$$

The one-to-one correspondence between the corners and the aforementioned subsets no longer holds when the object is positioned horizontally or vertically relative to the sensor surface coordinate. For these special cases, the corners take on extreme values in both the x and y directions, and the candidate positions for each corner are reorganized with other maximum or minimum operations in the other direction referred to the equation set 4.3.

$$
\begin{aligned}
S_{x_U y_U} &= X_U(S_{y_U}) \cup Y_U(S_{x_U}) \\
S_{x_L y_U} &= X_L(S_{y_U}) \cup Y_U(S_{x_L}) \\
S_{x_L y_L} &= X_L(S_{y_L}) \cup Y_L(S_{x_L}) \\
S_{x_U y_L} &= X_U(S_{y_L}) \cup Y_L(S_{x_U})
\end{aligned}
\tag{4.3}
$$

Once the subset corresponding to each corner is determined, all possible corner combinations can be enumerated. The proximity of the candidate corner coordinates to the actual object vertices can be assessed by the area of the quadrilateral formed by these points. The area is maximized when the four points are near the rectangle's vertices since this configuration provides the largest area for any set of four points constrained within a rectangular region. The quadrilaterals of different candidates in the example are represented by the shadings in Figure 4.2 and their areas can be determined using the MATLAB function **polyarea**(x, y), where x and y represent the values of the position set in the x and y directions. Finally, the candidate fiber positions that produce the maximum quadrilateral are identified as corners. The corresponding pseudocode is stated as follows.

Algorithm 1 Corner recognition algorithm

---

  function CORNER-RECOGNITION(S_B) **returns** fiber positions S_v close to object vertices
    inputs: S_B, coordinate set of receiving fibers (x_B, y_B) adjacent to object boundaries

    $\{S_1, S_2, S_3, S_4\} \leftarrow \{Y_U(S_B), X_L(S_B), Y_L(S_B), X_U(S_B)\}$
    if $\max(|S_1|, |S_2|, |S_3|, |S_4|) > 3$ **then**
        $S_1 \leftarrow X_U(S_1) \cup Y_U(S_4)$
        $S_2 \leftarrow X_L(S_1) \cup Y_U(S_2)$
        $S_3 \leftarrow X_L(S_3) \cup Y_L(S_2)$
        $S_4 \leftarrow X_U(S_3) \cup Y_L(S_4)$
    end if
    $A_{max} \leftarrow 0$
    $\{s_1, s_2, s_3, s_4\} \leftarrow \{1, 1, 1, 1\}$
    for $i = 1$ to $|S_1|$ **do**
        for $j = 1$ to $|S_2|$ **do**
            for $k = 1$ to $|S_3|$ **do**
                for $l = 1$ to $|S_4|$ **do**
                      $S_C \leftarrow \{S_1[i], S_2[j], S_3[k], S_4[l]\}$
                      $A_C \leftarrow$ **polyarea**$\big(S_C(x_B), S_C(y_B)\big)$
                      if $A_C > A_{max}$ **then**
                          $\{s_1, s_2, s_3, s_4\} \leftarrow \{i, j, k, l\}$
                          $A_{max} \leftarrow A_C$
                      end if
                end for
            end for
        end for
    end for
    $S_v \leftarrow \{S_1[s_1], S_2[s_2], S_3[s_3], S_4[s_4]\}$
    return S_v

The correspondence between the corner coordinates S_v and the object's vertices must also be established. The simplest approach is to arrange the coordinates based on the quadrant in which they are located within the object coordinate $x'O'y'$. According to the example in Figure 4.2, the vector from the origin $O'(\overline{x_B}, \overline{y_B})$ to a specific position $A(x_{v_1}, y_{v_1})$ in the global coordinate system xOy can be expressed as

$$\overrightarrow{O'A}(x, y) = \overrightarrow{OA}(x, y) - \overrightarrow{OO'}(x, y) = \begin{bmatrix} x_{v_1} - \overline{x_B} \\ y_{v_1} - \overline{y_B} \end{bmatrix} \tag{4.4}$$

The rotation angle $\alpha_{x'}$ of $x'O'y'$ can be roughly estimated by the slope of the line connecting the midpoints of the two closest pairs of corner positions. The transformed vector in the coordinate of the object is

$$\overrightarrow{O'A}(x', y') = \begin{bmatrix} \cos\alpha_{x'} & -\sin\alpha_{x'} \\ \sin\alpha_{x'} & \cos\alpha_{x'} \end{bmatrix} \cdot \overrightarrow{O'A}(x, y) \tag{4.5}$$

The signs of transformed coordinates in x' and y' directions return the quadrant that each vertex is located in. The vertices in the first to fourth quadrants are numbered 1 to 4, respectively, so that the top right vertex of the object always corresponds to the first corner in S_v, with $S_v[i] = (x_{v_i}, y_{v_i})$, $i \in \{1, 2, 3, 4\}$.

4.3. Edge classification

In this subsection, the index of the object boundary closest to each receiving fiber is figured out. This corresponds to the argument p of the relative angle θ that needs to be solved. Based on the positions of the fibers S_v close to the vertices, the set of positions S_B will be divided into four subsets E_1, E_2, E_3 and E_4, as illustrated in Figure 4.3.
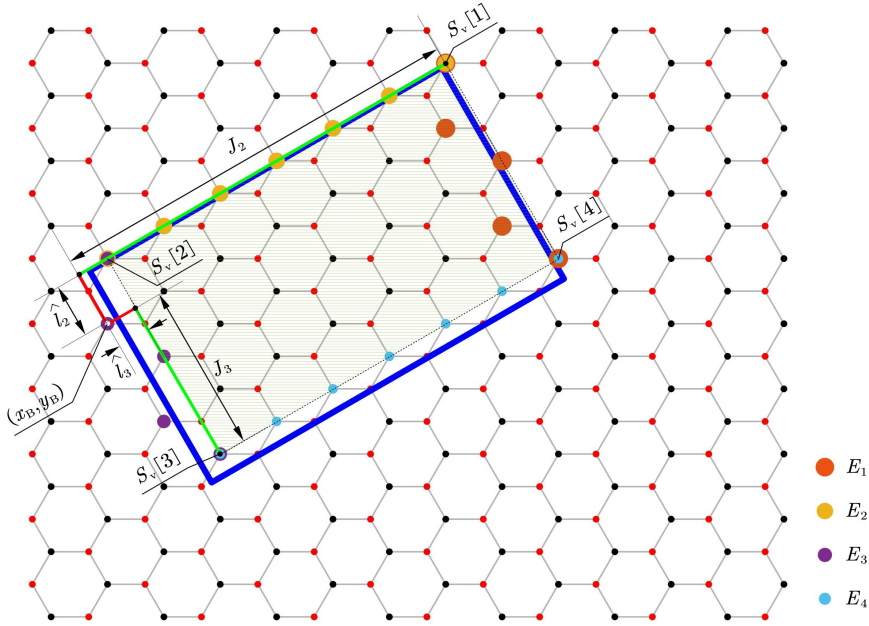
Figure 4.3: Edge classification based on identified vertices S_v

Each corner coordinate is first assigned to the position sets representing the two adjacent edges. The relations between edge indices and corner indices are stated as:

$$\begin{cases} E_m \cap E_n = S_v[m] \\ \{S_v[m], S_v[n]\} \subseteq E_n \end{cases} \quad m \in \{1, 2, 3, 4\}, \; n = \mathrm{mod}(m, 4) + 1 \tag{4.6}$$

The positions of the four corners determine the locations of the four edges and their expressions on the sensor surface.

$$\widehat{l}_n : y - y_{v_m} = \frac{y_{v_m} - y_{v_n}}{x_{v_m} - x_{v_n}} (x - x_{v_m}) \tag{4.7}$$

The formula for the distance from a point (x_B, y_B) to a line $ax + by + c = 0$ is given by

$$d(x_B, y_B) = \frac{|ax_B + by_B + c|}{\sqrt{a^2 + b^2}} \tag{4.8}$$

If the coefficients in the linear equation are normalized such that the square sum of coefficients in x and y terms equals 1, the distance can be directly expressed by the implicit function of the line $\widehat{l}(x, y) = ax + by + c$, i.e., $d(x_B, y_B) = |\widehat{l}(x_B, y_B)|$. In this case, the linear functions $\widehat{l}_n$ in Equation 4.7 are updated as

$$\widehat{l}_n(x, y) = \frac{y_{v_m} - y_{v_n}}{\|S_v[m] - S_v[n]\|}(x - x_{v_m}) - \frac{x_{v_m} - x_{v_n}}{\|S_v[m] - S_v[n]\|}(y - y_{v_m}) \tag{4.9}$$

According to the distances from fiber positions to various boundaries of the object, a new criterion is used to determine the nearest edge to the fiber. It is summarized as follows.

- **Criterion 1** : For any fiber position (x_B, y_B), the edge index with the smallest distance is denoted as p. If the distance from this fiber to any other edge differs from its distance to edge p by more than the inscribed radius of the hexagonal array r_c, then the fiber is assigned to edge E_p.

$$(x_B, y_B) \in S_B - S_v, \ p = \underset{n \in \{1,2,3,4\}}{\arg\min} \left|\widehat{l}_n(x_B, y_B)\right| \quad \forall n \neq p, \text{ s.t.}$$

$$\left|\widehat{l}_n(x_B, y_B)\right| - \left|\widehat{l}_p(x_B, y_B)\right| > r_c \Rightarrow (x_B, y_B) \in E_p$$

The above criterion applies to fibers that are far from the vertices. When the fiber is close to two edges of the object simultaneously, the estimated result based on the shortest distance may differ from the actual situation due to the rough initial solution of the object's vertices. However, for any fiber coordinates near a specific boundary E_n, their projected positions along the boundary should be restricted within the corners that define the boundary. This projected length is defined as the maximum distance from the projection point on the edge's line to the corresponding vertices and mathematically expressed as

$$J_n(x_B, y_B) = \max_k \sqrt{(x_{v_k} - x_B)^2 + (y_{v_k} - y_B)^2 - |\widehat{l}_n(x_B, y_B)|^2}, \ (x_{v_k}, y_{v_k}) \in S_v \cap E_n \tag{4.10}$$

The above parameter is utilized to check the feasibility of edge classification results with the following statement:

- **Criterion 2** : If the projected length from the fiber (x_B, y_B) to the edge E_n is greater than the distance between the two vertices that belong to the edge, then the fiber does not belong to the edge E_n.

$$\exists n \in \{1, 2, 3, 4\}, \ S_v[i], S_v[j] \in E_n \quad J_n(x_B, y_B) > \sqrt{(x_{v_i} - x_{v_j})^2 + (y_{v_i} - y_{v_j})^2} \Rightarrow (x_B, y_B) \notin E_n$$

The overall algorithm for classifying receiving fibers into distinct edges is stated in the following pseudocode, where the proximity is evaluated by the distance from the fiber coordinate to the edge. The boundary with the smallest distance to the fiber's position is returned, provided the projected fiber position on the boundary does not exceed the range spanned by the determined corner positions.

Algorithm 2 Edge classification

---

    function EDGE-CLASSIFICATION(S_B, S_v, r_c) **returns** fiber position sets E_1, E_2, E_3, E_4 close to each edge
        Inputs: S_B, coordinate set of receiving fibers (x_B, y_B) adjacent to object boundaries
                S_v, coordinates of fibers near object vertices determined by corner recognition algorithm
                r_c, inscribed radius of the hexagonal fiber array
        for $n = 1$ to 4 **do**
            $m \leftarrow \text{mod}(n + 2, 4) + 1$
            $L[n] \leftarrow \text{norm}(S_v[n], S_v[m])$
            $E[n] \leftarrow \{S_v[n], S_v[m]\}$
        end for
        for $i = 1$ to $|S_B|$ **do**
            $(x_B, y_B) \leftarrow S_B[i]$

$$p \leftarrow \underset{n}{\arg\min} |\widehat{l}_n(x_{\mathrm{B}}, y_{\mathrm{B}})|$$

$$q \leftarrow \underset{n \neq p}{\arg\min} |\widehat{l}_n(x_{\mathrm{B}}, y_{\mathrm{B}})|$$

if $|\widehat{l}_q(x_{\mathrm{B}}, y_{\mathrm{B}})| - |\widehat{l}_p(x_{\mathrm{B}}, y_{\mathrm{B}})| \leq r_{\mathrm{c}}$ **and** $S_{\mathrm{B}}[i]$ not in S_{v} **then**

    if $J_p(x_{\mathrm{B}}, y_{\mathrm{B}}) > L[p]$ **then**

        $E[q] \leftarrow \{E[q], (x_{\mathrm{B}}, y_{\mathrm{B}})\}$

    else

        $E[p] \leftarrow \{E[p], (x_{\mathrm{B}}, y_{\mathrm{B}})\}$

    end if

  else if $|\widehat{l}_q(x_{\mathrm{B}}, y_{\mathrm{B}})| - |\widehat{l}_p(x_{\mathrm{B}}, y_{\mathrm{B}})| > r_{\mathrm{c}}$ **then**

    $E[p] \leftarrow \{E[p], (x_{\mathrm{B}}, y_{\mathrm{B}})\}$

  end if

end for

return E

4.4. Local distance estimation

Once the input fiber coordinates are classified into distinct boundaries, the initial edge positions estimated by corners are replaced by the linear regression function of the fiber coordinates (x_{pq}, y_{pq}) in each subset E_p. The updated functions are expressed by

$$y - \overline{y}_p = \frac{\sum_q (x_{pq} - \overline{x_p})(y_{pq} - \overline{y_p})}{\sum_q (x_{pq} - \overline{x_p})^2}(x - \overline{x}_p) = \frac{C_{xy,p}}{C_{xx,p}}(x - \overline{x}_p) = \tan \widehat{\alpha_p}(x - \overline{x}_p), \quad q = 1, ..., |E_p| \tag{4.11}$$

The 2-argument arctangent function **arctan2** is applied[11], which can figure out the angle of a line segment in the Cartesian plane in the range $(-\pi, \pi]$ based on the quadrant in which one endpoint is located relative to the other endpoint. $\widehat{\alpha_p}$ are finally mapped to a range of $[0, 2\pi)$ and it can be stated as

$$\widehat{\alpha_p} = \begin{cases} \arctan2(C_{xy,p}, C_{xx,p}) & \arctan2(C_{xy,p}, C_{xx,p}) \geq 0 \\ \arctan2(C_{xy,p}, C_{xx,p}) + 2\pi & \arctan2(C_{xy,p}, C_{xx,p}) < 0 \end{cases} \tag{4.12}$$

The above formulas return four independent edge angles, while these arguments are indeed subject to geometric constraints where the adjacent edges are perpendicular, see Equation 2.8. These constraints, expressed in terms of the edge angles α_p, regulate the predicted results $\widehat{\alpha_p}$, to minimize the total difference. This leads to an optimized orientation α_0 of the object:

$$\alpha_0 = \underset{\alpha}{\arg\min} \sum_{p=1}^{4} \Delta^2(\alpha_p, \widehat{\alpha_p}) \quad \alpha \in [0, \pi) \tag{4.13}$$

$$\Delta(\alpha_p, \widehat{\alpha_p}) = \min(|\alpha_p - \widehat{\alpha_p}|, 2\pi - |\alpha_p - \widehat{\alpha_p}|) \tag{4.14}$$

The angular difference is the inferior angle formed by corresponding edges, which is always less than 180 degrees, see Equation 4.14. Equation 4.13 can be solved by the function **fminbnd** from the Optimization Toolbox in MATLAB. This function aims to find the input value at which the single variable function is minimized in a fixed interval. The arguments are stated as follows:

$$\alpha_0 = \textbf{fminbnd}(f_0, \mathrm{lb}, \mathrm{ub})$$

- α_0: design variable, the orientation of the object, dimension:[rad]
- f_0: objective function $f_0(\alpha)$ such that $f_0(\alpha_0) = \min f_0(\alpha)$, see Equation 4.13
- lb, ub: lower and upper bounds such that $\mathrm{lb} \leq \alpha_0 \leq \mathrm{ub}$, $\alpha_0 \in [0, \pi)$

This result, together with the average center of the fiber positions $(x_{\mathrm{B}}, y_{\mathrm{B}})$, forms an initial estimation of the planar DoFs of the measured object. In addition to the object orientation α_0 and the index of the nearest edge p, the orientation of the edge relative to the fiber transceivers in a unit cell also differs by the illumination case of transmitting fibers, with the index of illumination sequence denoted by subscript t. In this case, for any classified fiber position (x_{pq}, y_{pq}) with simulated sensor output N_{E}, the distance is predicted by

$$\widehat{d}_{pqt} = N_{\mathrm{F}}^{-1}\left(N = N_{\mathrm{E}}(x_{\mathrm{R}} = x_{pq}, y_{\mathrm{R}} = y_{pq}), \theta = \theta(\alpha_0, p, t)\right) \tag{4.15}$$

4.5. Global position estimation

In the previous section, the planar DoFs of the object were estimated based solely on the fiber positions. To enhance this estimation, we can incorporate the distance from each fiber to the edge, derived from the normalized light power. Specifically, by finding the optimal planar position that minimizes the discrepancy between the geometric distances from the fibers to the edges and the measured distances, we can achieve a more accurate estimate. The problem and its solution process are described in the following subsections.

4.5.1. Objective function

Combining Equations 2.15 and 4.8, the distance from a certain classified fiber position (x_{pq}, y_{pq}) to the corresponding edge of the object centered at (x, y) with orientation α is:

$$d_{pq}(x, y, \alpha) = \begin{cases} \sin\alpha(y_{pq} - y) + \cos\alpha(x_{pq} - x) - \frac{h}{2} & p = 1 \\ \cos\alpha(y_{pq} - y) - \sin\alpha(x_{pq} - x) - \frac{w}{2} & p = 2 \\ -\sin\alpha(y_{pq} - y) - \cos\alpha(x_{pq} - x) - \frac{h}{2} & p = 3 \\ -\cos\alpha(y_{pq} - y) + \sin\alpha(x_{pq} - x) - \frac{w}{2} & p = 4 \end{cases} \quad q = 1, ..., |E_p| \tag{4.16}$$

The operator of absolute value in Equation 4.8 is omitted and the negative results indicates that the fiber is inside the rectangular boundary. There always exists a planar position such that the difference between the local positions derived by the sensor output $\widehat{d}_{pqt}$ and by geometric relations d_{pq} is minimized. In this case, the total distance difference among the receiving fibers is considered an objective function and is expressed as:

$$f_{\mathrm{d}}(x, y, \alpha) = \sum_{\substack{p,q,t \\ (x_{pq}, y_{pq}) \in S_{\mathrm{B}} - S_{\mathrm{v}}}} \left(d_{pq}(x, y, \alpha) - \widehat{d}_{pqt} \right)^2 \tag{4.17}$$

To visualize the objective function, the distances $\widehat{d}_{pqt}$ are represented by a series of circles, where p, q indicates the center positions and distinct colors are used to distinguish the identified fibers from different illumination cases. The receiving fibers near the vertices are excluded since the generated light profile at these positions is not overlapped by a linear boundary. The minimized distance difference indicated by the objective function can also be stated as finding a planar position of the rectangular boundary such that it is tangential to most circles centered at the fiber positions, with radii determined by sensor output.
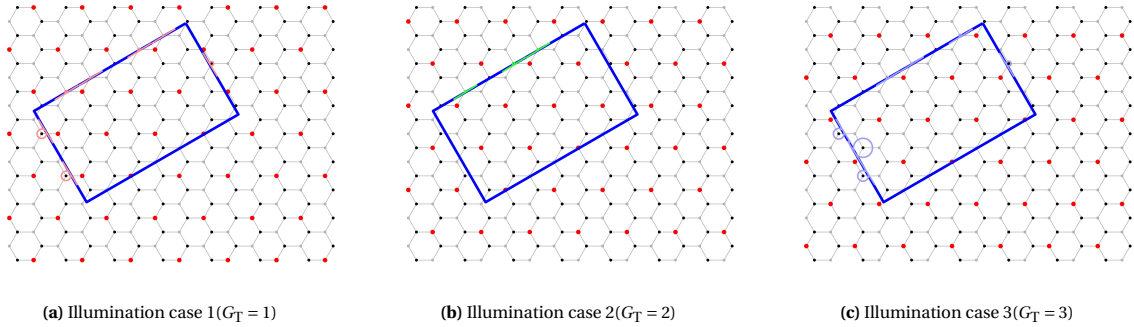

(a) Illumination case 1($G_{\mathrm{T}} = 1$)   (b) Illumination case 2($G_{\mathrm{T}} = 2$)   (c) Illumination case 3($G_{\mathrm{T}} = 3$)

Figure 4.4: Distance from receiving fibers to the boundaries of the object derived by the sensor output

4.5.2. Constraints

The selected receiving fibers could also return extreme light power for other illumination cases. Though these conditions can not determine the exact distance to the boundary, they do indicate a range where the edges might be. The corresponding inequalities will be set up as constraints to limit the feasible range of design variables.

According to the definition of transitional values of light power in Equation 2.26, the critical distance $\widehat{d}_{\mathrm{u}}$ when the fiber is fully covered by the object is:

$$\widehat{d}_{\mathrm{u}} = N_{\mathrm{F}}^{-1}\left(N = 0.95, \theta = \theta(\alpha_0, p, t)\right) \tag{4.18}$$

Similarly, the other critical value when the fiber is fully exposed is:

$$\widehat{d}_{\mathrm{l}} = N_{\mathrm{F}}^{-1}\left((N = 0.05, \theta = \theta(\alpha_0, p, t)\right) \tag{4.19}$$

The fitting result of light power model in Figure 2.8 indicates that the sensor output N decreases monotonically with the distance $\hat{d}_{pqt}$, and vice versa. Thus, for any fiber that returns $N > 0.95$ or $N < 0.05$, the constraints are proposed as

$$g_{\mathrm{u}}(x, y, \alpha) = d_{pq}(x, y, \alpha) - \widehat{d_{\mathrm{u}}} \leq 0 \quad N = N_{\mathrm{E}}(x_{\mathrm{R}} = x_{pq}, y_{\mathrm{R}} = y_{pq}) > 0.95 \tag{4.20}$$

$$g_{\mathrm{l}}(x, y, \alpha) = \widehat{d_{\mathrm{l}}} - d_{pq}(x, y, \alpha) \leq 0 \quad N = N_{\mathrm{E}}(x_{\mathrm{R}} = x_{pq}, y_{\mathrm{R}} = y_{pq}) < 0.05 \tag{4.21}$$

To visualize these constraints, circles are drawn for each illumination case, centered on the receiving fibers detected in the brightest and darkest conditions in Figure 4.5. The corresponding upper and lower distance bounds, $\widehat{d_{\mathrm{u}}}$ and $\widehat{d_{\mathrm{l}}}$, define the radii of these circles. In the brightest conditions, the solid line segments tangent to the circles represent the position constraints, and these segments must lie within the boundary of the object. In the darkest conditions, the dashed line segments represent the constraints, and they must fall outside the object boundary.
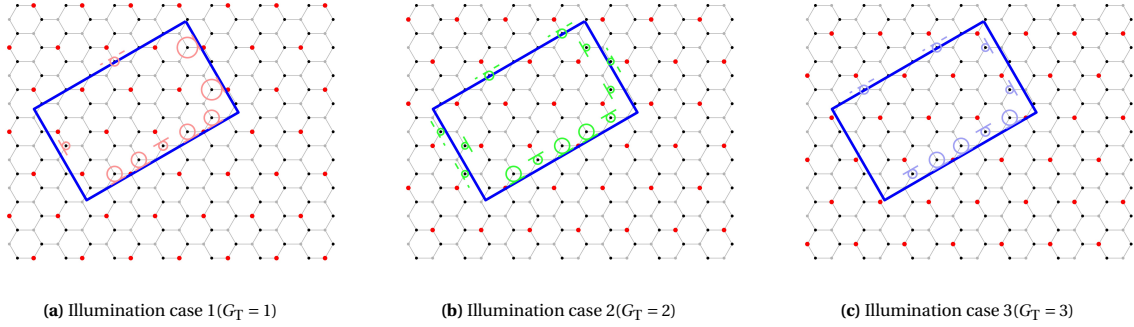

(a) Illumination case 1$(G_{\mathrm{T}} = 1)$ **(b)** Illumination case 2$(G_{\mathrm{T}} = 2)$ **(c)** Illumination case 3$(G_{\mathrm{T}} = 3)$

Figure 4.5: Constraints on feasible object positions

4.5.3. Iterative optimization

Overall, the estimation for the planar DoFs of the object can be described as the following mathematical form:

$$(x_{\mathrm{c}}, y_{\mathrm{c}}, \alpha_{\mathrm{c}}) = \underset{x, y, \alpha}{\operatorname{argmin}} f_{\mathrm{d}}(x, y, \alpha) \quad \text{s.t.} \quad \begin{cases} g_{\mathrm{u}}(x, y, \alpha) \leq 0 \\ g_{\mathrm{l}}(x, y, \alpha) \leq 0 \end{cases} \tag{4.22}$$

This problem can be solved by the function **fmincon** from Optimization Toolbox in MATLAB, which can find the minimum of the nonlinear multivariate function subject to linear or nonlinear constraints. The arguments for this function are listed as follows:

$$\mathbf{x} = \mathbf{fmincon}(f, \mathbf{x_0}, \mathbf{A}, \mathbf{b}, \mathbf{A_{eq}}, \mathbf{b_{eq}}, \mathbf{lb}, \mathbf{ub}, \mathbf{nonlcon})$$

- **x**: design variables, planar DoFs of the object $(x_{\mathrm{c}}, y_{\mathrm{c}}, \alpha_{\mathrm{c}})$
- f: objective function, $f_{\mathrm{d}}(x, y, \alpha)$ such that $f_{\mathrm{d}}(\mathbf{x}) = \min f_{\mathrm{d}}(x, y, \alpha)$
- **x$_0$**: initial estimation for design variables, $\mathbf{x_0} = (\overline{x_{\mathrm{B}}}, \overline{y_{\mathrm{B}}}, \alpha_0)$
- **A, b**: coefficients in linear inequality constraints $\mathbf{Ax} \leq \mathbf{b}$, empty for this optimization problem
- **A$_{eq}$, b$_{eq}$**: coefficients in linear equality constraints $\mathbf{A_{eq}} \mathbf{x} = \mathbf{b_{eq}}$, empty for this optimization problem
- **lb, ub**: lower and upper bounds such that $\mathbf{lb} \leq \mathbf{x} \leq \mathbf{ub}$, $x_{\mathrm{c}} \in [-105, 105]$, $y_{\mathrm{c}} \in [-74, 74]$, $\alpha_{\mathrm{c}} \in [0, \pi]$
- **nonlcon**: nonlinear inequality constraints such that **nonlcon**$(\mathbf{x}) \leq \mathbf{0}$, $\mathbf{nonlcon} = \begin{bmatrix} g_{\mathrm{u}}(\mathbf{x}) \\ g_{\mathrm{l}}(\mathbf{x}) \end{bmatrix}$

Noted that the constant terms $\hat{d}_{pqt}$, $\widehat{d_{\mathrm{u}}}$ and $\widehat{d_{\mathrm{l}}}$ predetermined in the objective function and constraints are related to α_0, the solution **x** is regarded as the new initial estimation **x$_0$** for the next optimization until the difference between two successive solutions α in the iteration process is less than 0.01 rad.

4.6. Results

In the given example, the estimated in-plane DoFs throughout the optimization process are detailed in Table 4.6. Initially, the estimation, represented by the magenta rectangle, is based solely on the positions of the receiving fibers located around the boundary. This estimation leads to an error in the center position with a millimeter-scale deviation since it merely averages the fiber positions. The limited information from the fiber positions also results in noticeable orientation discrepancies.

To improve the accuracy, theoretical relationships between the edge positions and the fiber outputs are incorporated into the optimization process, and the solutions rapidly converge towards the actual position, stabilizing within just two iterations. The final output from the proposed planar positioning algorithm, shown as a green rectangle, aligns almost perfectly with the true position, with a difference of less than 0.2 mm.
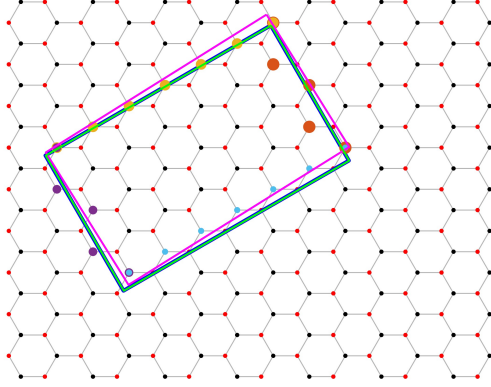
	x /mm	y /mm	α /rad
Real position	-20	12	0.5236
Initial estimation	-19.86	15.2	0.5582
Iteration 1	-19.87	11.98	0.5236
Iteration 2	-19.88	11.97	0.5236

Figure 4.6: Estimated positions in iterative process

With an interval of 30 degrees between 0 and 180 degrees, 10 positions are randomly selected for each angle, resulting in a total of 60 data sets. These samples are used to verify the accuracy and stability of the fiber array in planar measurements from a theoretical point of view, with the summary statistics for the error displayed as box plots in Figure 4.7. The lines inside each bin represent the median and the upper and lower edges of each bin represent the upper and lower quartiles, respectively. In the sample set, any result where at least one DoF exceeds the median of the corresponding dataset is highlighted, and the correlations between different DoFs are indicated with dashed lines.
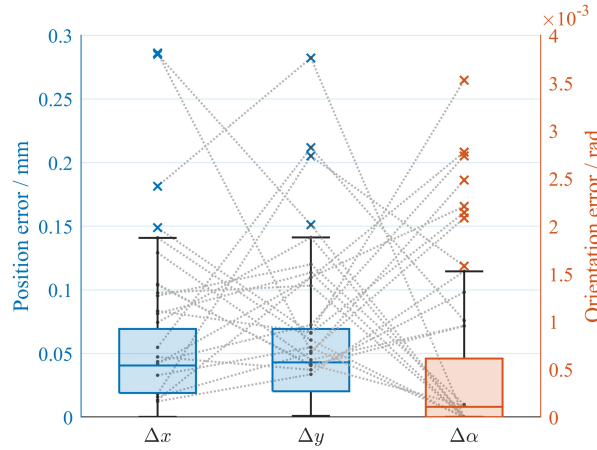

Figure 4.7: Error in planar DoFs by planar positioning algorithm ($r_c = 8$ mm, $r_0 = 7.2$ mm)

Overall, the median of error in the x and y directions of the sample is less than 0.05 mm, and the median angular error is close to 0.1 milli-rad. 6 out of 60 samples have an error greater than 0.15 mm along at least one

axis, which might be attributed to the limited number of detected receiving fibers that are used to construct the objective function in Section 4.5.1. In other words, The confidence level that the accuracy of the planar position measurement of the fiber array is within 0.15 mm is 90% over 60 randomly generated samples, and the maximum error does not exceed 0.3 mm.

4.7. Conclusion

This chapter introduces an algorithm designed to determine the planar degrees of freedom of an object using the positions of receiving fibers near its boundary and their outputs. Initially, fibers forming the largest quadrilateral are selected as corner points. Each remaining fiber is then assigned to the nearest edge of this quadrilateral, provided its position can be projected onto that edge. Linear fits are performed on the fiber groups to estimate the object's orientation. This preliminary estimate is refined by an optimization process that finds a planar position with the minimum sum of squared differences between the geometric and measured distances for all receiving fibers. The optimization process is iterated until the orientation stabilizes. The algorithm's performance is assessed using 60 test samples with arbitrary positions, yielding results where over 90% of the samples achieve a positional error of less than 0.15 mm in both the x and y directions.

5

Sensor surface construction

This chapter will discuss the design and fabrication of the sensor surface prototype incorporating multiple sets of fiber transceivers. Unlike the single-unit fiber transceiver prototype discussed in Chapter 3, the primary challenge in constructing the sensor surface lies in the alignment and arrangement of the fibers, light sources, and sensing system. Following the flow chart in Figure 5.1, the required number of LEDs for each channel is first determined. The illumination control module developed in the previous chapter is then adapted to manage these LED groups within the sensor surface prototype. Next, the design and manufacturing methods for the critical components used for fiber alignment are explained. Finally, the components of the sensor surface prototype are assembled, and the image readout from the bottom-view CCD camera is examined. This readout will be used to evaluate the sensor's performance in the following chapter.
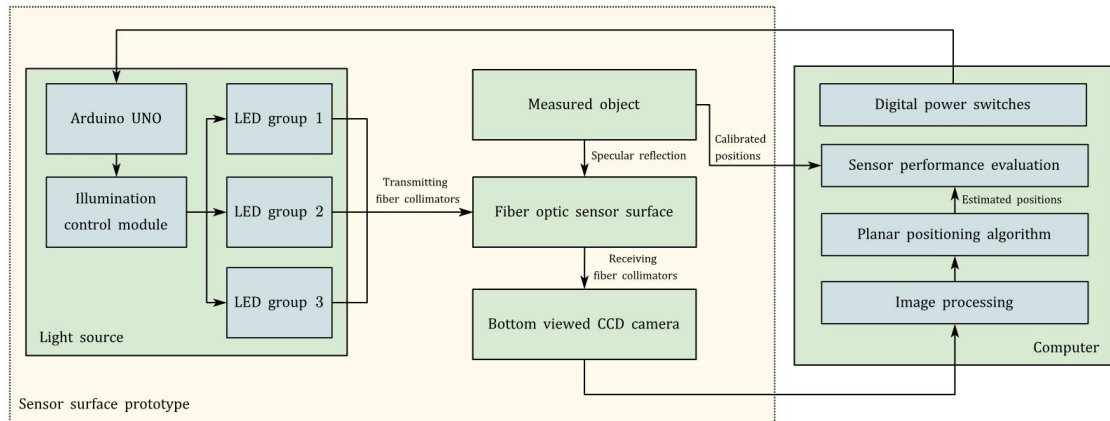

Figure 5.1: Overview of planar positioning system by distributed fiber optic sensors

5.1. LED array configuration

In the experiments described in Chapter 3, each transmitting fiber was paired with an individual light source. However, as the fiber array expanded, the drawbacks of this layout, such as increased energy and material consumption, as well as more complex control, became increasingly apparent. To effectively reduce the number of light sources and simplify the system layout, in the construction of the sensing surface, each LED simultaneously illuminates all the transmitting fibers in the same row. The fiber layout on the A5-sized plane is shown in the array in Figure 4.6. Considering the installation margin of the sensing surface in practical applications, the fibers in the upper and lower rows have been omitted. The sensing array is ultimately divided into 17 rows, with each row containing 7 transmitting fibers and 7 receiving fibers. Three sets of light sources are illuminated sequentially, and each set requires at least 6 LED lamps.

The circuit diagram for one of the LED groups is illustrated in Figure 5.2. Compared to the common-emitter

amplifier circuit in Figure 3.3, the voltage of the digital output pins V_{cc} no longer serves as a power supply, as it cannot provide sufficient voltage when the number of loads increases. Instead, V_{cc} now controls the relay in the transistor's collector branch, with the switch closing the circuit when the coil is energized. The power supply has been replaced by a 9V lithium battery ($V_{sup} = 9$ V), and six LED lamps are evenly distributed across two branches in the circuit.
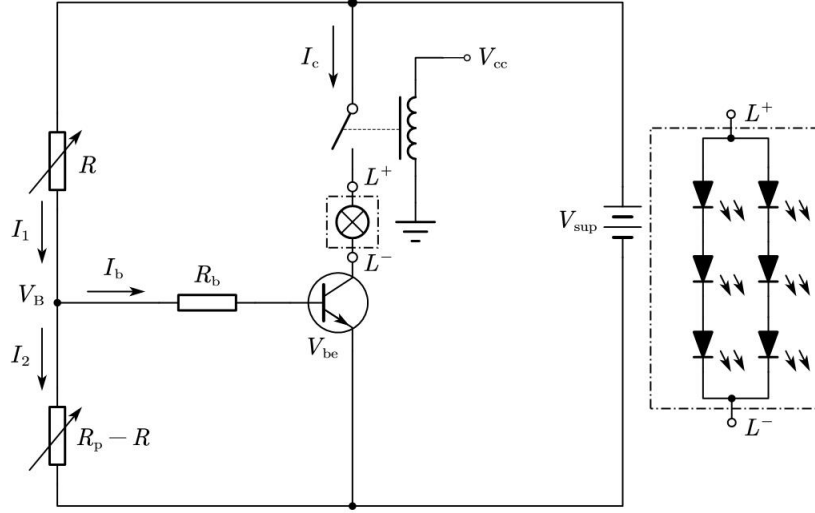

Figure 5.2: Illumination control module for LED array in the sensor surface

Equations 3.8 and 3.10 are used to verify the design when the loads and power supplies are altered. The base resistor R_b and the total resistance of the potentiometer R_p are set to the same values as in the previous experiment. In these expressions, the critical base current is calculated as

$$I_b \left(R = \frac{V_{be}}{V_{sup}} R_p \right) = \frac{V_{sup} - 2V_{be}}{R_b + R_p \left(\frac{V_{be}}{V_{sup}} - \left(\frac{V_{be}}{V_{sup}} \right)^2 \right)} = 2.38 \text{ mA} \tag{5.1}$$

For two branches in the LED array, the nominated and maximum allowable summed current in the collector end are

$$I_{cs,nom} = 2I_{c,nom} = 40 \text{ mA} \tag{5.2}$$

$$I_{cs,max} = 2I_{c,max} = 100 \text{ mA} \tag{5.3}$$

Based on the provided data sheet[48], the DC gain β_{cs} drops to 30 when the collector current increases to 100 mA. In this case, the upper and lower bounds of the base current from Equation 3.8 and 3.10 are updated as

$$I_{b,u} = \frac{I_{cs,max}}{\beta_{cs}} = 3.33 \text{ mA} \tag{5.4}$$

$$I_{b,l} = \frac{2I_{cs,nom}}{\beta} = 1.33 \text{ mA} \tag{5.5}$$

where the critical base current satisfies these constraints. Thus, the above configuration of light sources will be adopted in the construction of the sensor surface.

5.2. Prototype part design

The design of the parts to be manufactured and an overview of their assembly in the sensor surface prototype are shown in Figure 5.3. The overall frame of the prototype is divided into two layers. The LED groups and other necessary electronic components are placed on the bottom layer of the frame, with their positions secured by an additional circuit board holder. The sensor surface is located on the upper layer of the frame and primarily consists of a fiber fixing plate, a light diffusion sheet, and spacers that separate different planes.

The holes for securing the optical fibers are arranged in a honeycomb pattern on the fiber fixing plate. The light emitted from different groups of transmitting fibers is distinguished by colored inverted cones. The other ends of these fibers are connected to the corresponding light sources via transmitting fiber collimators, based on their group and row positions. The receiving fiber collimator gathers all the receiving fibers into the scope of the CCD camera. During the measurement, the real position of the reflective object placed above the sensor surface is fixed by a position calibration board. The detailed design and manufacturing methods of the key parts will be introduced in the following subsections.
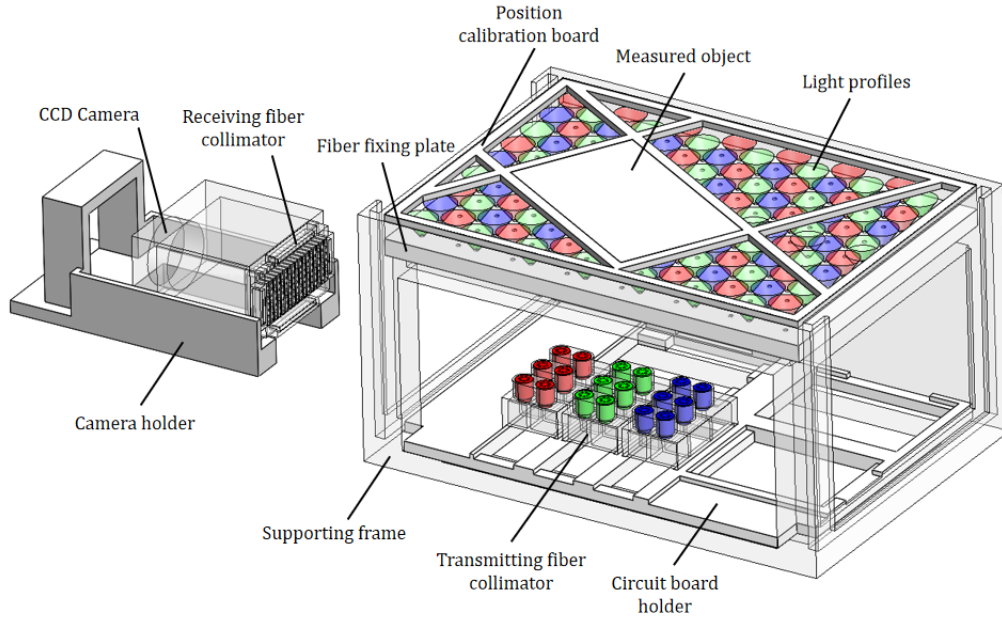

Figure 5.3: SOLIDWORKS models for sensor surface prototype

5.2.1. Transmitting fiber collimator

The transmitting fiber collimator consists of six cylindrical elements that hold the transmitting fibers and a light source housing. From the perspective view in Figure 5.4a, the interior of the light source housing consists of six separate cavities. The light source array is partitioned within the housing to prevent interference from other light sources. The protruding section at the bottom of the housing aligns with the grooves in the circuit board holder to ensure that the position of the transmitting fibers relative to the light sources remains fixed.
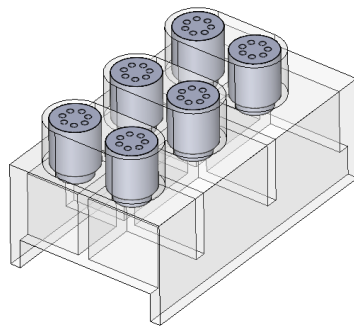
(a) Perspective view
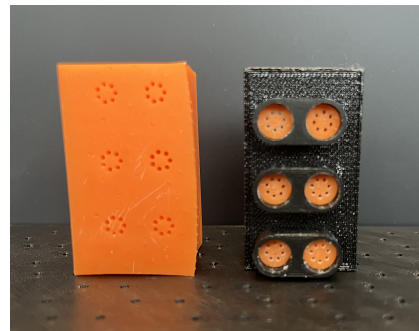
(b) Physical models

Figure 5.4: Design and fabrication of transmitting fiber collimator

Given the 1 mm diameter of the fiber cores, the MSLA 3D printer was used to fabricate the fiber holders to ensure precise assembly. However, MSLA printers often produce noticeable surface deformation in large

flat parts, as illustrated by the component on the left in Figure 5.4b. To address this, the mounting holes are separated from the housing. For the housing, an FDM 3D printer was used to avoid deformation. The opaque material from the FDM printer also effectively blocks most ambient light, unlike the translucent orange part produced by the MSLA printer.
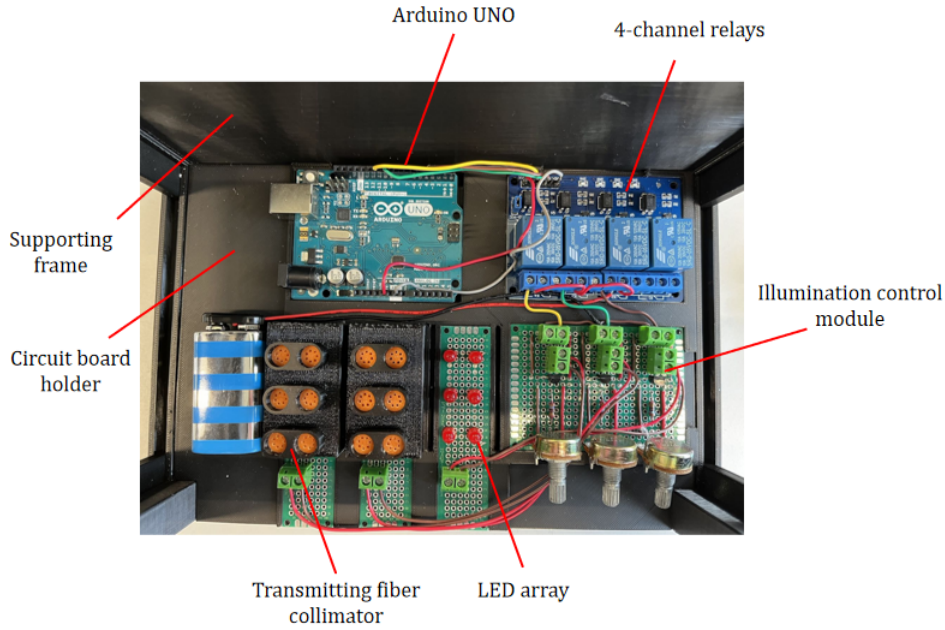

Figure 5.5: Assembly of components at the bottom layer of the sensor surface prototype

The physical layout of all components on the bottom layer of the sensor surface prototype is shown in Figure 5.5. The supporting frame and circuit board holder, which support the electronic components, are manufactured using an FDM printer. To ensure proper fiber alignment, the flat-bottomed LEDs are soldered onto the circuit board with their central axes aligned with the 7 mounting holes positioned directly above.

5.2.2. Receiving fiber collimator

The detailed structure of the receiving fiber collimator is shown in the perspective view in Figure 5.6a. The components that align the cores of the receiving fibers consist of a series of stepped sheets. These parts are mounted on a collimator holder with corresponding grooves. The light blocker secures both the camera lens and the collimator holder at its ends, effectively preventing external light interference while maintaining the fixed relative positions of the fibers and the lens.
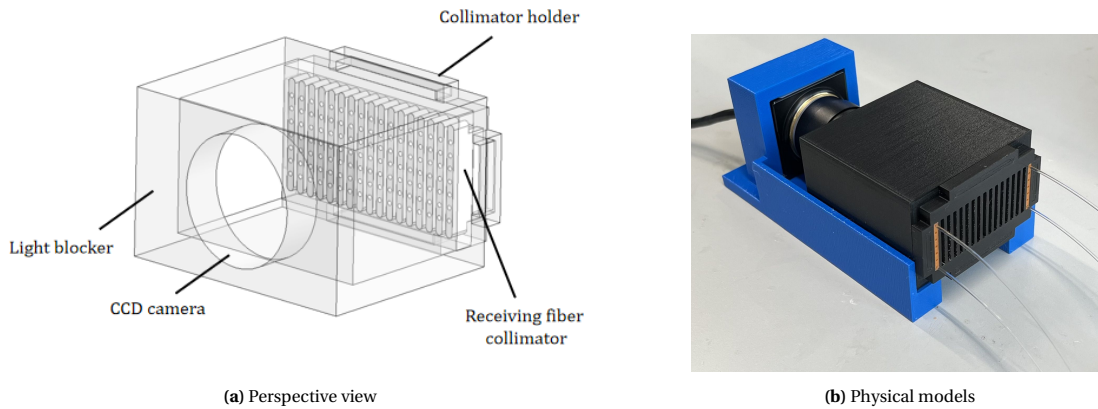

(a) Perspective view   **(b)** Physical models

Figure 5.6: Design and fabrication of receiving fiber collimator

Similar to the manufacturing process of the transmitting fiber collimator, the fiber fixing sheets are made by an MSLA printer, while the other components are produced with an FDM printer, as shown in the physical models in Figure 5.6b. The camera and receiving fiber collimator are both mounted on the camera holder to minimize the impact of ground disturbances on the measurements. Four receiving fibers are installed at the corner positions of the fiber array to verify the coverage of the fibers within the camera view. The result in Figure 5.7 shows that the CCD camera can detect all 17 columns and 7 rows of receiving fibers when they are aligned by the fiber collimator within a 50×24 mm^2 area, positioned 45 mm from the camera lens.

Figure 5.7: Camera view

5.2.3. Sensor surface

The fiber fixing plate is the fundamental layer in the sensor surface, where the positions of the transmitting and receiving fibers are secured. Given the sufficiently dispersed distribution of the fibers among the sensor surface, the inner diameter of the fixing holes is matched to the diameter of the jacketed fibers. For large components with regular shapes and high precision requirements, additive manufacturing methods such as FDM printing can face issues such as long processing times, poor flatness, and inconsistent hole diameters, see the plate on the right in Figure 5.8. In contrast, using a laser cutter to create a hole array in a blank acrylic plate reduces the processing time from 1 day to 15 minutes. Furthermore, hole diameters are uniform, with maximum processing errors below 0.2 mm. The laser-cutted sample with a series of hole diameters is illustrated on the left, where the 2.2 mm diameter hole provides the best fit with the jacketed fiber.
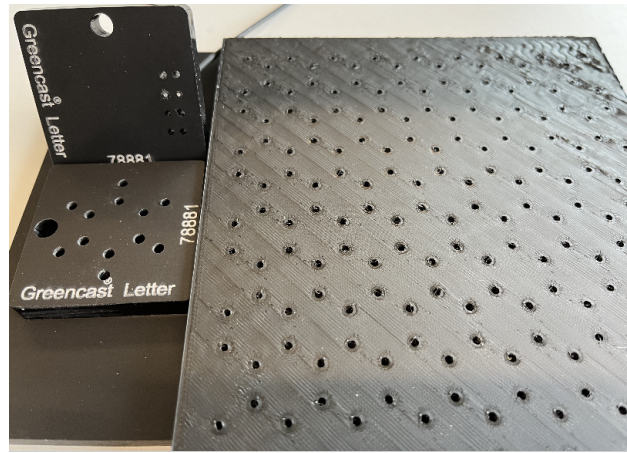

Figure 5.8: Fabrication of fiber fixing plate

The overall structure of the sensing surface is shown in Figure 5.9. The layers are numbered and their detailed information is tabulated. To maintain the same distances between the ends of the optical fibers, the light diffusion sheet, and the object's surface as in the experiments in Chapter 3, 5 mm and 3 mm thick transparent PMMA plates are added between the fiber fixing plate and the light diffusion sheet, and between the light diffusion sheet and the measured object, respectively.

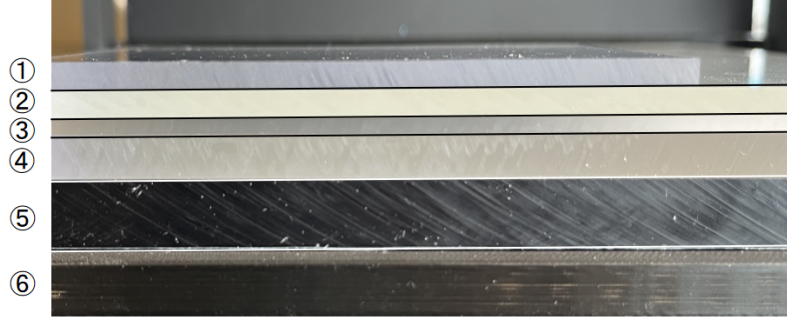| Layer number	Name	Material	Thickness(mm)	Size(mm^2)
1	Reflective object	Acrylic	3	100×60
2	Transparent spacer	PMMA	3	210×148
3	Light diffusion sheet PyraLED YT290	Acrylic	1.8	210×148
4	Transparent spacer	PMMA	5	210×148
5	Fiber fixing plate	Acrylic	8	210×148
6	Supporting frame	PLA	-	-

Figure 5.9: Cross section view of sensor surface

5.3. Fiber installation

This section primarily introduces the distribution of fibers on the sensor surface and their connection to the fiber collimators. The receiving fibers are numbered according to their row and column indices in a rectangular array from the receiving fiber collimator. The other ends of the receiving fibers are connected to holes in the sensor surface that are excluded from the colored circles illustrated in Figure 5.10. These positions, within the honeycomb grid, can be divided into 17 rows in the xy plane, with each row containing 7 positions. For each receiving fiber, the column index U from left to right at the end mounted in the receiving fiber collimator corresponds to the row index from top to bottom at the other end on the sensor surface. Alternatively, the column index of a receiving fiber from left to right in a row on the sensor surface corresponds to its row index V from top to bottom in the fiber collimator.
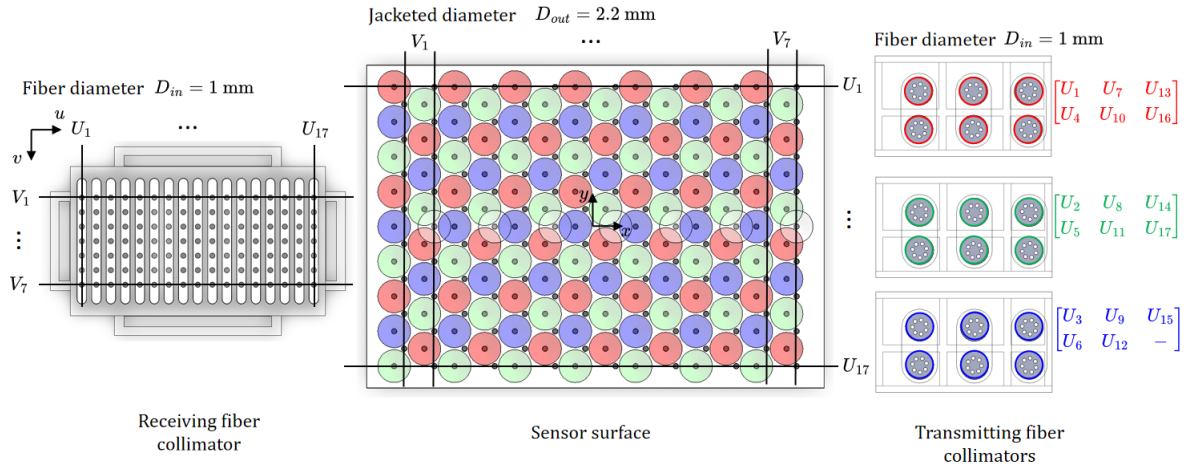

Figure 5.10: Connections between the fiber collimators and the sensor surface

Given the in-radius of honeycomb grid r_c as well as the column and row indices of a receiving fiber U, V, the

mapped coordinate (x, y) on the sensor surface can be calculated as

$$(x, y) = T_s(U, V)$$

$$x = \begin{cases} \dfrac{r_c}{\sqrt{3}} (6(V-4)+2) & \text{odd } U \\ \dfrac{r_c}{\sqrt{3}} (6(V-4)-1) & \text{even } U \end{cases}$$

$$y = -r_c(U-9)$$

(5.6)

The transmitting fibers on the sensor surface are marked by different colored circles. They are also arranged in 17 rows, each row containing 7 positions. The positions within each row are indicated by colored row indices. These position sets are assigned to 3×2 matrices representing individual light sources under each transmitting fiber collimator.

The physical setup, with the fiber collimators connected to the sensor surface according to the layout described above, is shown in Figure 5.11. The transmitting fibers are fully enclosed within the supporting frame, making them less susceptible to ambient light interference and easier to shield. To save installation space, their outer jackets have been completely removed. In contrast, the receiving fibers extend out from the sensor surface, so the outer jackets on the non-mating portions are retained to reduce ambient light interference. The yellow rubber sleeves used to fit the fibers and the sensor surface in Figure 5.11c are stripped from the insulation layer of 18 AWG single-strand tinned copper wire. The inner and outer diameters of these sleeves match those of the fiber jackets, but they have a lower off-center rate and exhibit less radial deformation compared to the original fiber jackets. The dark cores in the sensing array represent the transmitting fibers, while the light cores represent the receiving fibers when the receiving fiber collimator is exposed to ambient light and the light source is turned off.
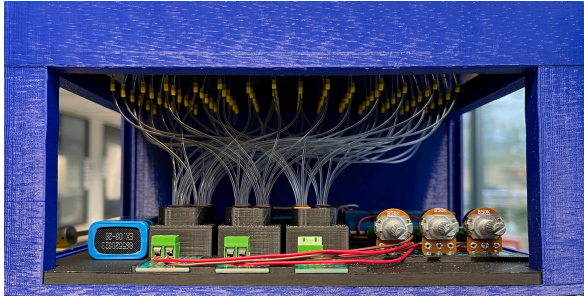
(a) Transmitting fibers
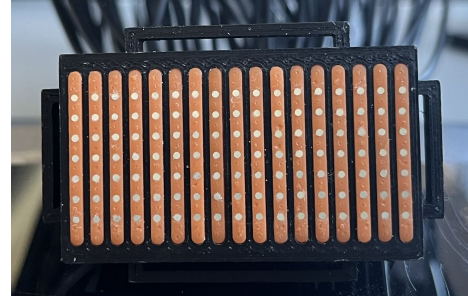
(b) Receiving fibers
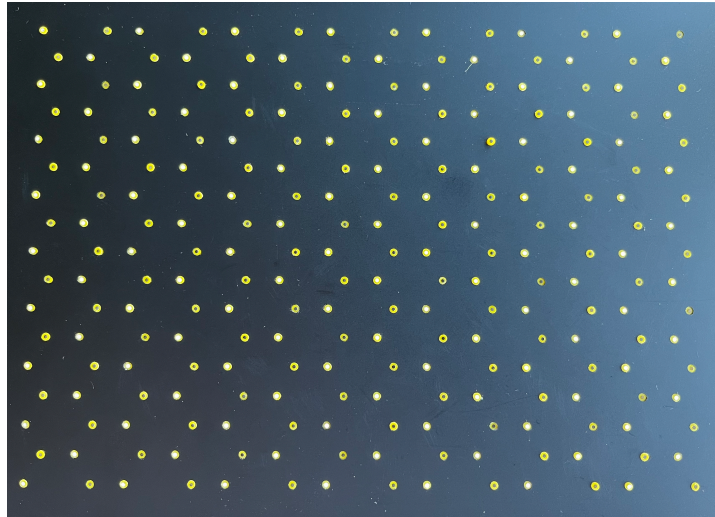
(c) Sensor surface

Figure 5.11: Physical setup of optical fibers in the sensor surface prototype

5.4. Setup evaluation

The overall assembly diagram of the sensor surface prototype is shown in Figure 5.12. Multiple sets of position calibration boards were 3D printed to secure the test objects at various positions on the sensor surface, allowing for the evaluation of the measurement accuracy of the sensor surface prototype. During the position measurement, an additional A5-sized black acrylic plate behind the sensor surface in the figure will be placed over the sensor surface to block ambient light.
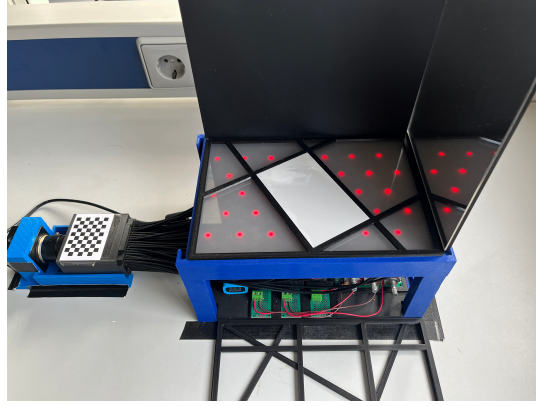

Figure 5.12: Overall setup of the sensor surface prototype

Some inherent parameters of the sensor surface prototype are determined before conducting the position measurement and the performance of the overall setup is evaluated by the maximum brightness differences of the receiving fibers. As illustrated in Figure 5.13, the brightest receiving fibers are obtained when the sensor surface is covered by an A5-sized mirror, while the lowest brightness is returned when the black acrylic sheet is placed directly on the sensor surface. Following the fiber handling method mentioned in subsection 3.1.1, homogeneous light spots without cracks can be observed in most of the receiving fibers for the brightest case. However, there is a significant brightness variation among the receiving fibers, which may be due to differences in the bending of the transmitting fibers during assembly, leading to varying levels of loss in light propagation.

(a) Brightest case($G_T = 1$) **(b)** Brightest case($G_T = 2$) **(c)** Brightest case($G_T = 3$)

(d) Darkest case($G_T = 1$) **(e)** Darkest case($G_T = 2$) **(f)** Darkest case($G_T = 3$)
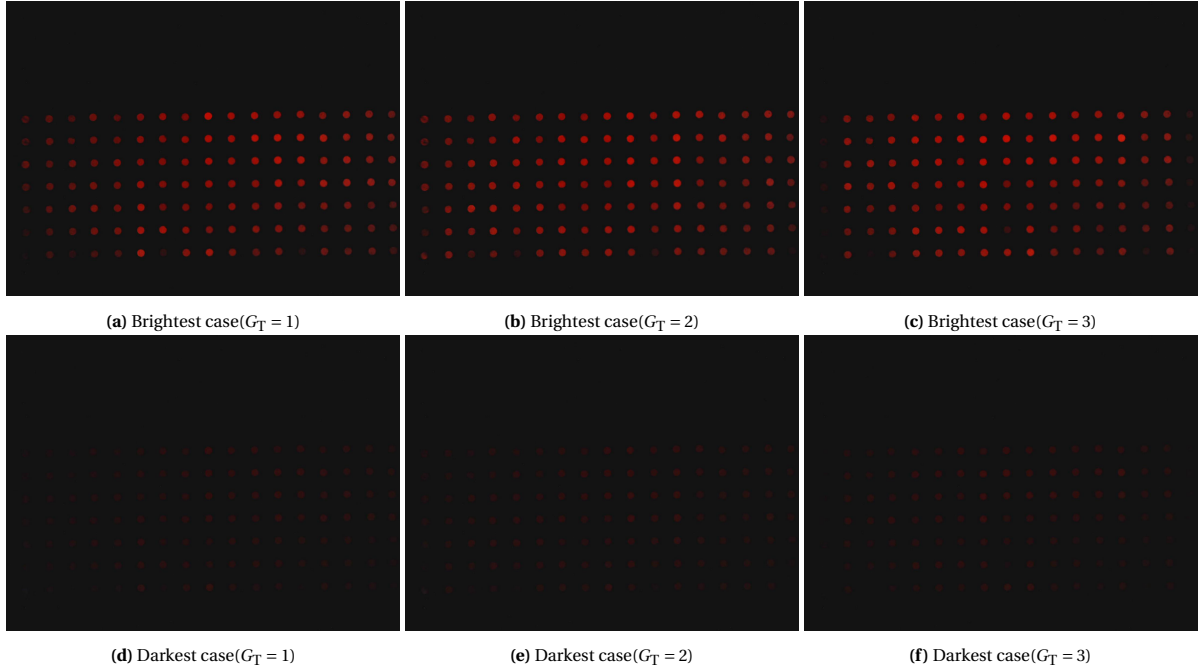
Figure 5.13: Maximum intensity difference among receiving fibers

The average R-value for each receiving fiber is recorded for both the brightest and darkest conditions. The distribution of these maximum brightness differences is presented in the form of a histogram in Figure 5.14. Fibers with a brightness difference of less than 30 are less distinguishable. Their boundaries are hardly detected by image processing algorithms even under the brightest conditions, making these fibers unusable in the corresponding illumination case. It is noted that the number of unusable fibers is significantly higher when the transmitting fibers in group 3 are illuminated compared to other illumination cases. Most of these unusable fibers are located at the boundary positions of the fiber array on the sensor surface, where there are no adjacent illuminated transmitting fibers.

For the remaining usable receiving fibers, their brightness differences are mainly distributed between 70 and 100, with medians close to 90, similar to the brightness differences returned by a single set of fibers from the experiment in Chapter 3. Despite the large deviation in brightness differences, the detected brightness of the receiving fibers will be linearly normalized based on their maximum and minimum brightness values during actual measurements. This process helps reduce the impact of variations in light propagation loss on the measurement. Additionally, other interfering factors, such as light directly reflected from the surface of the light diffusion sheet, can be offset through this normalization.
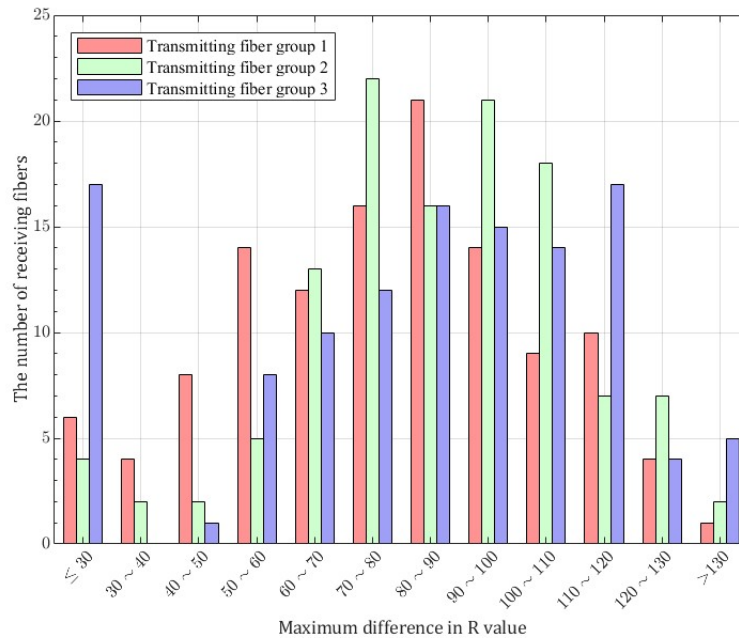

Figure 5.14: Histogram of maximum intensity differences among receiving fibers in different illumination cases

5.5. Conclusion

This chapter primarily discusses the detailed steps of the prototype for distributed fiber optic sensors, covering structural design, manufacturing methods, and practical implementation. The light sources of the prototype are divided into three groups, each consisting of six LEDs. Relays are introduced as digital switches for each group and these light sources are powered by an additional battery with a higher voltage to ensure sufficient brightness. The light sources and electronic components are mounted on the bottom layer of the prototype, while the sensor surface with the transmitting and receiving fibers is on the upper layer. The other end of these transmitting fibers is aligned with light sources in rows using collimators, and receiving fibers are similarly grouped and focused in the camera's field of view. In the prototype, the mounting holes on the fiber-fixing plates in the sensor surface are crafted through laser cutting, while the components used to hold the fibers are manufactured using MSLA printers, and the remaining parts are produced by FDM printers. The preliminary calibration results of the prototype show that the light sources provide sufficient brightness for receiving fibers, while there is a great variation in brightness differences due to the discrepancies in the bending of transmitting fibers. As a result, the measured brightness of a light spot is required to be linearly normalized based on their extreme values for each receiving fiber.

6

Experimental validations

In the previous chapter, the prototype of the distributed fiber optic sensing system was established, which encodes the object's position information into images. To complete the system, the proposed planar positioning algorithm must be integrated to decode the position information from these images via a computer. This chapter first introduces the process of deriving the data needed for position measurement from the raw images. The established data, along with the key steps of the planar positioning method, form a complete algorithm that returns the planar DoFs of the measured object. Various position samples are evaluated to validate the system's design, and potential errors are analyzed.
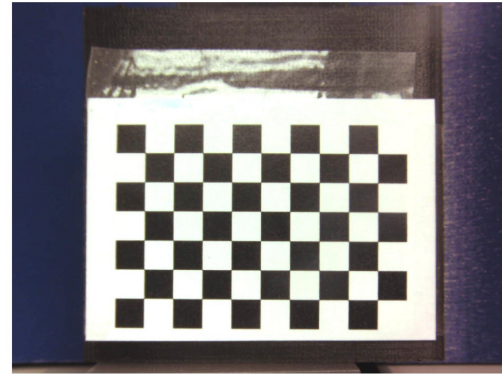
6.1. Data preprocessing

6.1.1. Image undistortion

When using a CCD camera to capture images of objects at close range, distortion often occurs at the edges of the image, causing what should be straight lines to appear unnaturally deformed, as shown in Figure 6.1a. Moreover, the greater the magnification of the object within the camera's field of view, the more severe the distortion. This not only affects the contours of the fibers being measured but also hinders the calibration of the relative positions of the receiving fibers within the array. To ensure the quality of the images, the camera's intrinsic parameters must be determined to correct the captured images before using the CCD camera.

<div align="center">
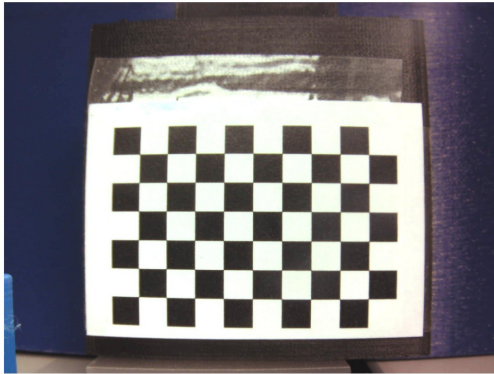
(a) Snapshot by CCD camera before image undistortion   (b) Snapshot by CCD camera after image undistortion

Figure 6.1: Distortion in the camera view and its correction

</div>

An asymmetric checkerboard with a square size of 5×5 mm^2 was used as the calibration board for image undistortion. Among 10 distorted image samples, the calibration board was positioned at various angles and distances relative to the camera lens. These samples were uniformly processed using the MATLAB ap-

plication Camera Calibrator, which returned the camera's intrinsic parameters. As shown in Figure 6.1b, the transformed image illustrates the correction of previously distorted patterns.

6.1.2. Fiber array localization

In distributed fiber optic sensors, each receiving fiber contains two types of positional information. The fiber's center and radius within the camera's field of view are used to evaluate the fiber's output $\widehat{P}$, while its absolute position on the sensor surface serves as an input for the planar positioning algorithm to estimate the orientation of the measured object. Since the position of the fibers remains fixed, the positional information will be directly applied as inherent parameters in subsequent measurements once the fibers are localized. The mapping between the two ends of each receiving fiber has been established by its row and column indices in the array from the camera's view in Equation 5.6, while the image processing function returns the absolute position of the receiving fibers in the real measurements. Therefore, these unordered absolute positions need to be rearranged into a matrix to localize the fibers more effectively.
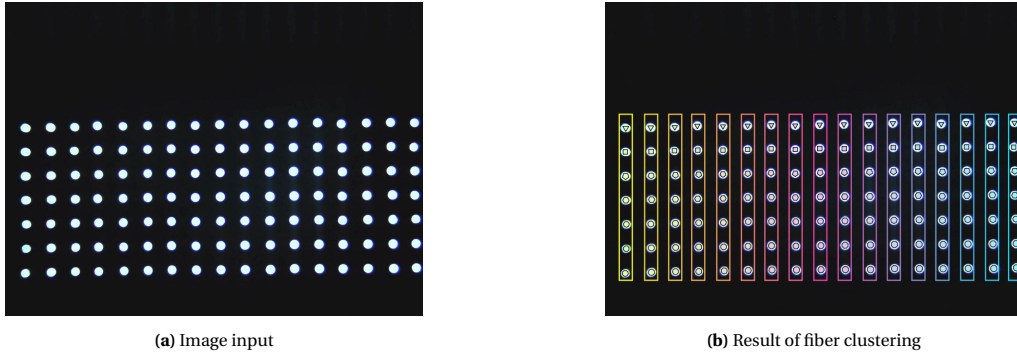

(a) Image input   **(b)** Result of fiber clustering

Figure 6.2: Localization of fiber positions in the image frame

Figure 6.2a illustrates the distribution of receiving fibers in the undistorted image when the sensor surface is uncovered. Although the centers of the receiving fibers in each row or column may not be perfectly aligned, they can still be accurately grouped into distinct regions. To achieve this, the fiber positions are divided into 7 row clusters and 17 column clusters using MATLAB's **kmeans**(dataset, cluster number) function. This function provides the cluster index for each observation along with the centroid positions of the clusters. The centroid positions are then sorted in ascending order for both directions, and the cluster indices are updated based on this sorted ranking.

To visualize the result of fiber clustering, different colored bounding boxes and various shapes of regular polygons are used to distinguish fibers located in different columns or rows, as shown in Figure 6.2b. As a result, the calibrated center position of a receiving fiber $(\widehat{u_R}, \widehat{v_R})$ with a radius $\widehat{r_R}$ can also be accessed by its column and row indices U and V, and the mapping achieved by the above process is denoted as:

$$(\widehat{u_R}, \widehat{v_R}, \widehat{r_R}) = T_c(\text{image}, U, V) \tag{6.1}$$

6.1.3. Database setup

In the simulation platform, many parameters for determining the normalized light power N are directly given, such as positions and maximum brightness differences of receiving fibers. However, in the physical prototype, these parameters need to be calibrated through image input, which involves a series of intermediate variables and processing functions, as listed in Table 6.1. The constructed distributed fiber optic sensor consists of 17 rows and 7 columns of receiving fibers, with each receiving fiber corresponding to 3 different illumination cases G_T. If the listed parameters are stored for all cases in a table, it would result in a 357×14 table, containing significant redundant information. To achieve more efficient data storage and retrieval, the parameters are divided into three separate databases based on the images serving as their information source. These databases are linked through the shared row and column indices, U and V.

The first database contains all parameters related to the positional information of the receiving fibers, with the corresponding table denoted as $D_X = D_X(U, V, \widehat{u_R}, \widehat{v_R}, \widehat{r_R}, x_R, y_R)$. These parameters are not affected by the actual position of the measured object as well as the illumination cases, so they are listed separately.

Table 6.1: Stored parameters and related processing functions for a receiving fiber in the prototype

Item	Definition	Processing function	Image input	Ref.
U	Column index in camera frame	-	-	-
V	Row index in camera frame	-	-	-
G_T	Illumination case	-	-	-
$(\widehat{u_R}, \widehat{v_R}, \widehat{r_R})$	Position and size in camera frame	$T_c(\text{image}, U, V)$	Figure 6.2a	Equation 6.1
(x_R, y_R)	Position on sensor surface	$T_s(U, V)$	-	Equation 5.6
G_R	Group number	$\mod(U-1,3)+1$	-	Equation 2.2
$\widehat{P_L}$	Minimum brightness	$\widehat{P}(\text{image}, \widehat{u_R}, \widehat{v_R}, \widehat{r_R})$	Figure 5.13a-5.13c	Equation 3.12
$\widehat{P_U}$	Maximum brightness	$\widehat{P}(\text{image}, \widehat{u_R}, \widehat{v_R}, \widehat{r_R})$	Figure 5.13d-5.13f	Equation 3.12
t	Position index	$T_G(G_T, G_R)$	-	Equation 2.4
$\widehat{P}$	Brightness	$\widehat{P}(\text{image}, \widehat{u_R}, \widehat{v_R}, \widehat{r_R})$	Figure 6.3d-6.3f	Equation 3.12
$\widehat{N}$	Normalized brightness	$\dfrac{\widehat{P} - \widehat{P_L}}{\widehat{P_U} - \widehat{P_L}}$	-	Equation 2.20

The second database, $D_P = D_P(U, V, G_T, G_R, \widehat{P_L}, \widehat{P_U})$, stores the extreme brightness values for all illumination cases. For brightness evaluation, positional information is directly retrieved from the first database rather than through a repeated fiber localization process on the image. This method will also be used when evaluating the brightness of fibers in images where the object is placed above the sensor surface. Since this database is independent of the measured object's position, just like the first database, both databases remain consistent across different measurement samples.

(a) Prototype setup($G_T = 1$)   **(b)** Prototype setup($G_T = 2$)   **(c)** Prototype setup($G_T = 3$)

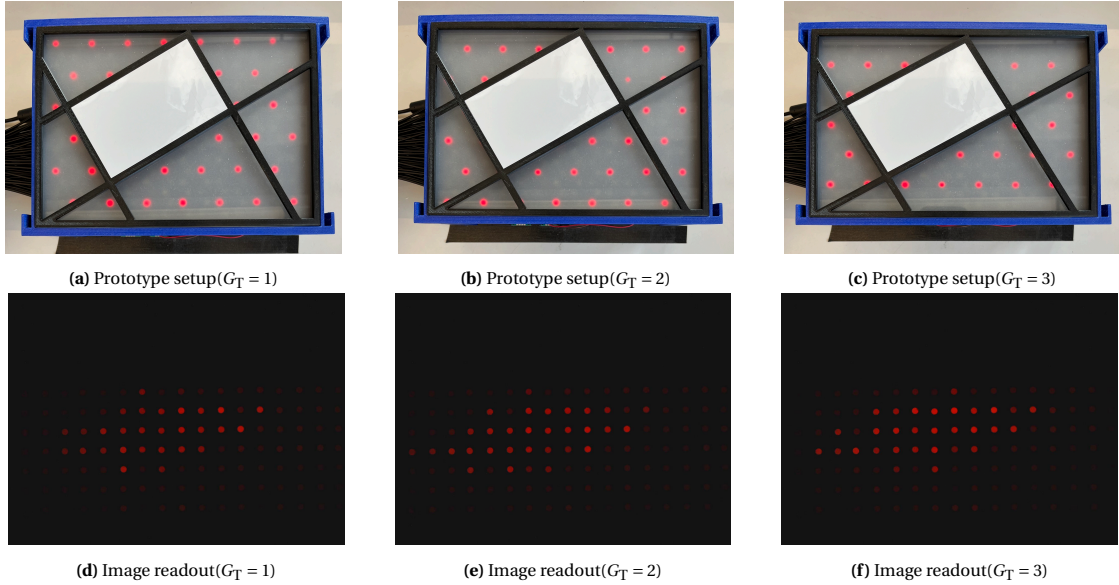**(d)** Image readout($G_T = 1$)   **(e)** Image readout($G_T = 2$)   **(f)** Image readout($G_T = 3$)

Figure 6.3: Prototype layout and related image inputs in different illumination cases used for position measurement

Figure 6.3 shows the observed light spots of the receiving fibers when the reflective object is placed on the sensor surface. In reality, the illustrated setup of the prototype is entirely covered by a black acrylic plate to block the ambient light. Each receiving fiber generates three distinct inputs when illuminated by different groups of transmitting fibers, and the brightness varies with the object's position. Therefore, the parameters derived from these images are assigned to a separate database $D_M(U, V, G_T, t, \widehat{P}, \widehat{N})$. As the measured brightness from the fibers with a maximum difference below 30 is not reliable, the database is filtered by the following expression:

$$D_M = D_M \left\{ [U \ V \ G_T] \in D_p \{ \widehat{P_U} - \widehat{P_L} \geq 30, [U \ V \ G_T] \}, [U \ V \ G_T \ t \ \widehat{P} \ \widehat{N}] \right\} \tag{6.2}$$

When using curly braces to represent a database, the first item within the braces is a condition that identifies

all rows in the table that meet this criterion, while the second item specifies the columns selected from the table.

6.2. Boundary fiber selection

This section explains how to extract the positions of fibers near the boundary of the measured object from image inputs. In the physical prototype, the normalized brightness is no longer used directly to determine whether a fiber is near the boundary. Instead, the positions of all illuminated fibers are identified and the fibers located on the outermost side are selected. This approach reduces the impact of brightness fluctuations in the brightest case on the boundary detection during actual measurements.

Based on the image inputs in Figure 6.3, row and column indices are selected from the database D_M where the normalized brightness at those locations exceeds a specified lower threshold for any illumination case. This process can be stated in the following mathematical form:

$$\widehat{S}_I = D_M \left\{ \widehat{N} \geq 0.1, [U \ V \ G_T] \right\} \tag{6.3}$$

$$\widehat{X}_I = D_X \left\{ [U \ V] \in \widehat{S}_I(U,V), [x_R \ y_R] \right\} \tag{6.4}$$

(a) Image readout($G_T = 1$) **(b)** Image readout($G_T = 2$) **(c)** Image readout($G_T = 3$)
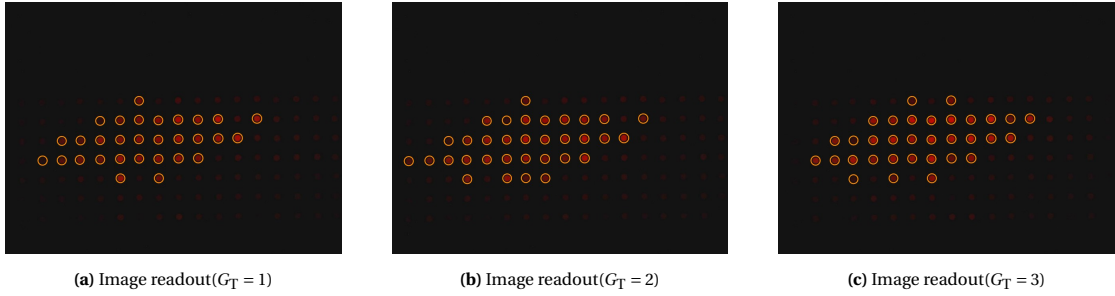
Figure 6.4: Recognition of illuminated receiving fibers

The result of the preliminary selection of receiving fibers is illustrated in Figure 6.4. In these images, the identified receiving fibers are marked by orange circles. They are clustered in the same region for different illumination cases, with slight variations in their positions near the contour. These positions are united and plotted from the perspective of the sensor surface in Figure 6.5a. It can be seen that the identified fibers are all constrained within the rectangular boundary representing the actual object's location and its vicinity.
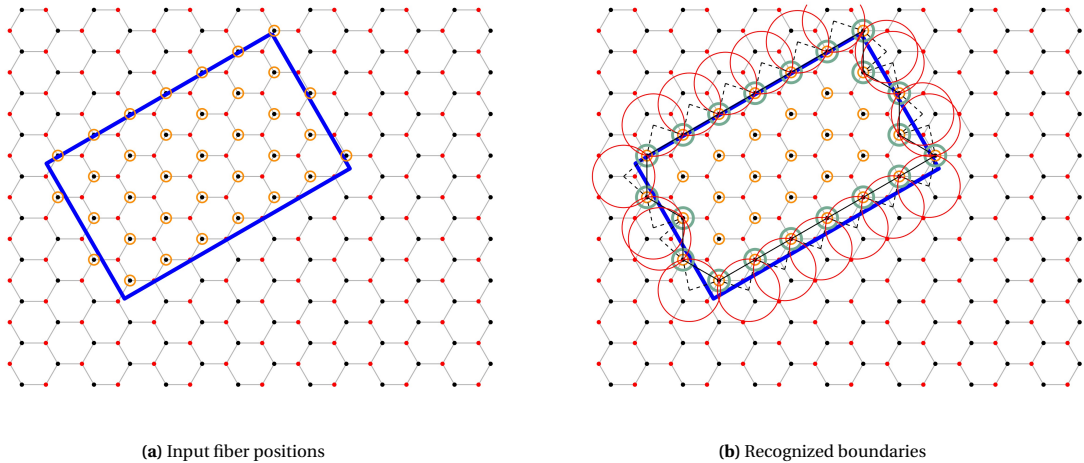

(a) Input fiber positions **(b)** Recognized boundaries

Figure 6.5: Ball-pivoting algorithm[19] for determining the boundary sets

Fibers near the object boundary and located at the outer part of the position set are distributed at the vertices of a concave polygon, which can encompass all the fiber positions. Based on existing geometric processing

methods[19], the ball-pivoting algorithm is applied to identify fibers in the boundary profile. Following open source programming, the algorithm constructs a ball with radius r_b, which starts by approaching the point with the smallest y value from below and coincides with that point. The ball is then rotated counterclockwise around this point until its boundary coincides with another point in the set. The rotation center is updated to the new coinciding point and the process is repeated until a previously encountered point is reached. The trajectory of the ball in the given example is illustrated in Figure 6.5b and all the encountered points $\widehat{X_B}$ in the position sets $\widehat{X_I}$ are returned, with the processing function defined by:

$$\widehat{X_B} = \textbf{ballconcave}(\widehat{X_I}, r_b) \tag{6.5}$$

The radius r_b used in the algorithm must be appropriately constrained. If r_b is too small, it may not form a closed contour; if it is too large, some boundary points might be missed. Specifically, r_b should be large enough so that the longest edge of the formed polygon is smaller than the ball's diameter but not so large that the ball's radius exceeds the distance when it intersects three points in the set simultaneously. Given that the distance between adjacent receiving fibers and the minimum diameter when three fibers are located on the same circle simultaneously are $2r_c$ and $4r_c$, respectively, an average radius is taken for r_b, with $r_b = \frac{3}{2}r_c = 12$ mm.

6.3. Algorithm integration

In the previous sections, the data inputs needed for the position recognition algorithm have been extracted from the raw image information. The main steps of the planar positioning algorithm and their implementation methods have been explained in detail in Chapter 4. Based on the database measured from the prototype, the steps, and their associated parameters are integrated to form a complete algorithm, which takes databases from different images as input and outputs planar DoFs.

Algorithm 3 Programming overview for distributed fiber optic sensors with inputs provided by prototype

---

    function PLANAR-POSITIONING(D_M) **returns** planar position of the measured object (x_c, y_c, α_c)
    Inputs: D_M, database including the brightness of all receiving fibers for all illumination cases
    Local variables: (x, y, α), planar DoFs concerning the geometric center and orientation of the sensor surface
                    f_0, objective function for initially estimated orientation α_0, formulated by α and E
                    $d_{pq}(x, y, \alpha)$, geometric distance from the fiber to the object positioned at (x, y, α)
                    $\theta(\alpha_0, p, t)$, relative angle between the edge and the illuminated transmitting fiber
    Intrinsic parameters: r_b, ball radius for ball-pivoting algorithm
                    r_c, in-radius of the hexagon in fiber array
                    D_X, database including all positional information of all receiving fibers
                    N_F^{-1}, fitted model between normalized brightness and local distance
                    (h, w), length and width of the measured object, applied in function d_{pq}

    $\widehat{S_I} \leftarrow D_M\{\widehat{N} \geq 0.1, [U\ V\ G_T]\}$
    $\widehat{X_I} \leftarrow D_X\{[U\ V] \in \widehat{S_I}(U, V), [x_R\ y_R]\}$
    $\widehat{X_B} \leftarrow \textbf{ballconcave}(\widehat{X_I}, r_b)$
    $\widehat{X_v} \leftarrow \textbf{corner-recognition}(\widehat{X_B})$
    $E \leftarrow \textbf{edge-classification}(\widehat{X_B}, \widehat{X_v}, r_c)$
    $\alpha_0 \leftarrow \textbf{fminbnd}(f_0, 0, \pi)$
    $x_0 \leftarrow [\widehat{X_B}(x_R)\ \widehat{X_B}(y_R)\ \alpha_0]$
    converge $\leftarrow$ **False**
    while NOT converge **do**
        $f_d(x, y, \alpha) \leftarrow 0$
        $g(x, y, \alpha) \leftarrow 0$
        for $p = 1$ to $|E|$ **do**
            $E_p \leftarrow E[p]$
            for $q = 1$ to $|E_p|$ **do**
                if $E_p[q] \notin \widehat{X_v}$ **then**
                    $(x_{pq}, y_{pq}) \leftarrow E_p[q]$
                    $[U_{pq}\ V_{pq}] \leftarrow D_X\{(x_R, y_R) = (x_{pq}, y_{pq}), [U\ V]\}$

for $k = 1$ to 3 **do**
$\quad$ $[t_{pq}\ \widehat{N}_{pq}] \leftarrow D_{\mathrm{M}}\big\{(U,V,G_{\mathrm{T}}) = (U_{pq}, V_{pq}, k),\ [t\ \widehat{N}]\big\}$
$\quad$ $\theta_{pq} \leftarrow \theta(\alpha_0, p, t_{pq})$
$\quad$ **if** $\widehat{N}_{pq} \geq 0.9$ **then**
$\quad\quad$ $\widehat{d} \leftarrow N_{\mathrm{F}}^{-1}(0.9, \theta_{pq})$
$\quad\quad$ $g(x,y,\alpha) \leftarrow [g(x,y,\alpha)\ d_{pq}(x,y,\alpha) - \widehat{d}]$
$\quad$ **else if** $\widehat{N}_{pq} \leq 0.1$ **then**
$\quad\quad$ $\widehat{d} \leftarrow N_{\mathrm{F}}^{-1}(0.1, \theta_{pq})$
$\quad\quad$ $g(x,y,\alpha) \leftarrow [g(x,y,\alpha)\ \widehat{d} - d_{pq}(x,y,\alpha)]$
$\quad$ **else**
$\quad\quad$ $\widehat{d} \leftarrow N_{\mathrm{F}}^{-1}(\widehat{N}_{pq}, \theta_{pq})$
$\quad\quad$ $f_d(x,y,\alpha) \leftarrow f_d(x,y,\alpha) + \big(\widehat{d} - d_{pq}(x,y,\alpha)\big)^2$
$\quad$ **end if**
end for
end if
end for
end for
$[x_{\mathrm{c}}\ y_{\mathrm{c}}\ \alpha_{\mathrm{c}}] \leftarrow$ **fmincon**$(f = f_d(x,y,\alpha),\ \mathbf{x_0} = x_0,\ \mathbf{lb} = [-105\ -74\ 0],\ \mathbf{ub} = [105\ 74\ \pi],\ \mathbf{nonlcon} = g(x,y,\alpha))$
if $|\alpha_{\mathrm{c}} - \alpha_0| \leq 0.01$ **then**
$\quad$ converge $\leftarrow$ **True**
else
$\quad$ $x_0 \leftarrow [x_{\mathrm{c}}\ y_{\mathrm{c}}\ \alpha_{\mathrm{c}}]$
$\quad$ $\alpha_0 \leftarrow \alpha_{\mathrm{c}}$
end if
end while
return $x_{\mathrm{c}}, y_{\mathrm{c}}, \alpha_{\mathrm{c}}$

The preliminary results obtained from the input data based on image information are shown in Figure 6.6. Compared to the results obtained from the simulation in Chapter 4, both share the same initial solution, as indicated by the magenta rectangular frame, due to the identical identification of boundary fiber positions. The orientation optimized through iterative adjustments based on the actual normalized brightness closely aligns with the actual angle. While the accuracy of the center position estimation is inferior to the simulation, the errors in both the horizontal and vertical directions are ultimately within 1 mm, achieving sub-millimeter measurement accuracy in this example.
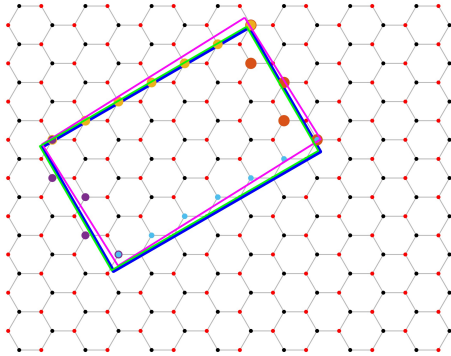
	x /mm	y /mm	α /rad
Real position	-20	12	0.5236
Initial estimation	-19.86	15.20	0.5582
Iteration 1	-20.66	12.44	0.5275
Iteration 2	-20.83	12.35	0.5240

Figure 6.6: Iterative optimization in planar position estimation based on image input provided by prototype

6.4. System evaluation

A single set of position data alone cannot fully reflect the measurement performance of the prototype. To avoid the contingency of experimental results, four additional position calibration boards were produced to place the measured object at various locations on the sensor surface, with different rotation angles.

The physical configurations of the position calibration boards are illustrated in Figure 6.7. Each calibration board can provide four test positions by rotating and flipping it. This results in a total of 20 measurement samples, covering most of the receiving fibers on the sensor surface. Calibration 1 has been illustrated in the example given in Figure 6.3. Calibration boards 2 and 5 are designed to verify the accuracy of the measurement when the object is placed horizontally or vertically. For calibration board 3, the object is positioned at a random angle(13°) to observe the algorithm's compatibility under irregular orientations. Calibration board 4 corresponds to the position of the object where the maximum error is returned among the simulated samples.

(a) Calibration board 2

(b) Calibration board 3

(c) Calibration board 4

(d) Calibration board 5
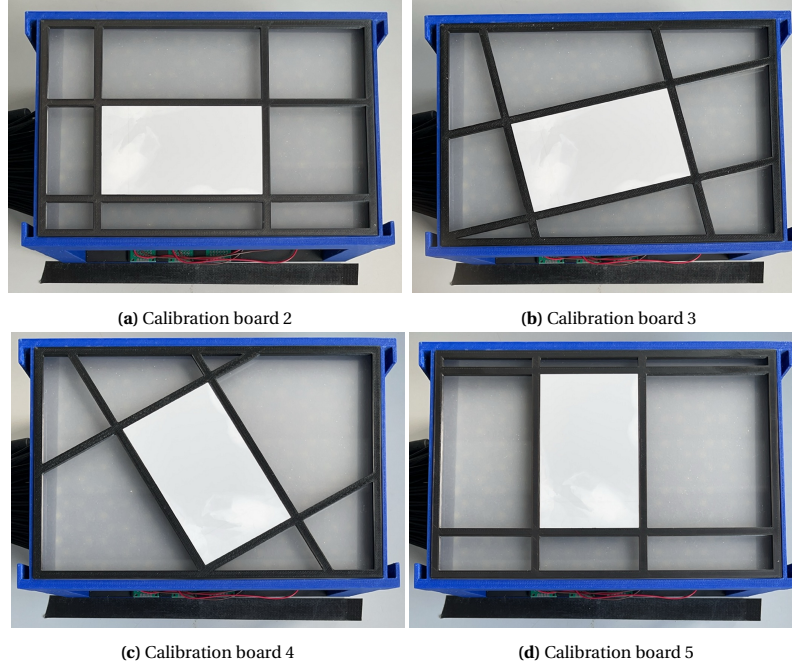
Figure 6.7: Position measurement samples for distributed fiber optic sensors

For each measurement sample, the steps outlined in the previous section are repeated. The raw image inputs and measurement results are detailed in Appendix B. The errors corresponding to each position sample are listed in Table 6.2 and are summarized in a box plot in Figure 6.8. To provide a clearer depiction of the actual angular errors, the units in the chart are expressed in degrees.

Overall, compared to the simulation results shown in Figure 4.7, the actual errors are almost magnified by an order of magnitude. The median errors among the 20 test samples are 0.77 mm and 0.65 mm, respectively. However, compared to previous studies, these experimental errors are now on the same order of magnitude as the average simulation error (0.35 mm for 30 covered receiving fibers in Figure 2.1a), and the sensor configuration developed in this research also achieves an average angular estimation error of less than 1 degree.
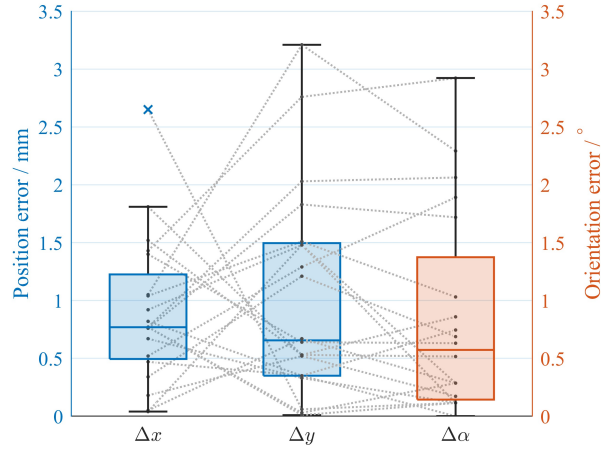
Figure 6.8: Error in planar DoFs by sensor prototype

When using the calibration board as a reference, there are some variations in the results across different sets of measurement data. When the object is placed horizontally or vertically, the angle error returned by the sensing system is significantly smaller compared to other sample sets, and in one of the DoFs, the estimation results are noticeably better than the average. However, in the other direction, occasional outliers may appear, such as Δx in No.5.

For the remaining three sets of calibration boards, the overall error in any degree of freedom gradually increases from board 1 to 3 and then to 4. The calibration board with the largest error in the simulation also shows the largest error in the actual measurements. Irregular angles do not significantly affect the position estimation in real measurements, but some cases of unreasonable fiber classification locally (see Table B.2) can lead to an increase of angular error on a scale of 0.5 to 1 degree.

Table 6.2: Measurement error for each sample

No.	Board	Real position			Error		
		x_0/mm	y_0/mm	α/°	Δx/mm	Δy/mm	$\Delta \alpha$/°
1		-20	12	30	0.82	0.35	0
2	1	20	-12	30	1.81	0.53	0.516
3		20	12	150	1.40	0.64	0.631
4		-20	-12	150	0.47	0.35	0.745
5		-16	-16	0	2.65	0.06	0.115
6	2	16	16	0	0.77	0.01	0.115
7		16	-16	0	0.05	0.67	0.286
8		-16	16	0	0.77	0.03	0.286
9		-6	-12	13	1.04	1.51	1.031
10	3	6	12	13	1.52	0.53	0.859
11		6	-12	167	0.34	1.21	0.688
12		-6	12	167	0.52	1.29	1.891
13		-0.5	-7.4	120	1.43	2.76	2.922
14	4	0.5	7.4	120	0.76	1.83	1.719
15		0.5	-7.4	60	0.77	2.03	2.063
16		-0.5	7.4	60	1.05	3.21	2.292
17		-10	8	90	0.92	1.48	0.115
18	5	10	-8	90	0.18	0.52	0.172
19		10	8	90	0.04	1.48	0.286
20		-10	-8	90	0.67	0.33	0.115

6.5. Error analysis

The planar DoFs of the object are determined by the brightness of the receiving fibers and their position distribution on the sensing plane. Therefore, the errors are mainly analyzed from these two aspects. Other possible sources of error in the developed prototype are listed as follows. The impact of these factors is relatively minor and they will not be explained in detail.

- Positioning accuracy of the position calibration board
- Position accuracy of the receiving fibers on the sensor surface
- Localization of receiving fibers in the camera frame
- Instability of the light source's brightness
- Surface quality of receiving fibers and measured object

6.5.1. Brightness loss

The horizontal segments in Figure 6.9 illustrates the deviation between the actual distance d_{pq} from each receiving fiber to the object boundary and the distance $\hat{d}_{pqt}$ derived from the measured brightness $\hat{N}$ based on the fitted model N_{F}. The ideal brightness for each receiving fiber in the fitted model is indicated by a star marker, with the vertical dashed lines representing the difference in brightness between the ideal and actual conditions.
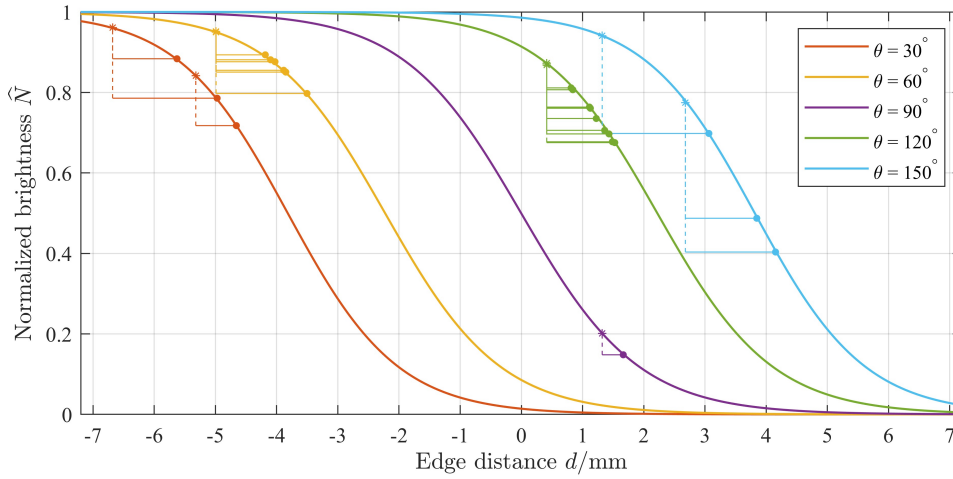
Figure 6.9: Influence of brightness loss on position estimation(Sample No.1)

The result for the given example shows that many data points appearing at different positions on the curve in actual measurements theoretically correspond to the same position, and the actual brightness at these points is lower than the ideal brightness to varying degrees. These phenomena are sufficient to indicate that, in addition to the fitting errors inherent to the function itself, brightness loss originating from different fibers can also lead to shifts in position estimation. Moreover, this loss cannot be fully compensated by the linear normalization of brightness.

6.5.2. Anisotropic fiber spacing

In a hexagonal array, the line intervals between receiving fibers vary when evaluated from different angles, which could also affect the sensitivity of position measurement. As shown in Figure 6.10, the fibers are most densely packed at 0, 60, and 120 degrees, while they are most sparsely packed at 30, 90, and 150 degrees. This means that when one side of the rectangular object being measured aligns with the dense direction, its adjacent side will align with the sparse direction.
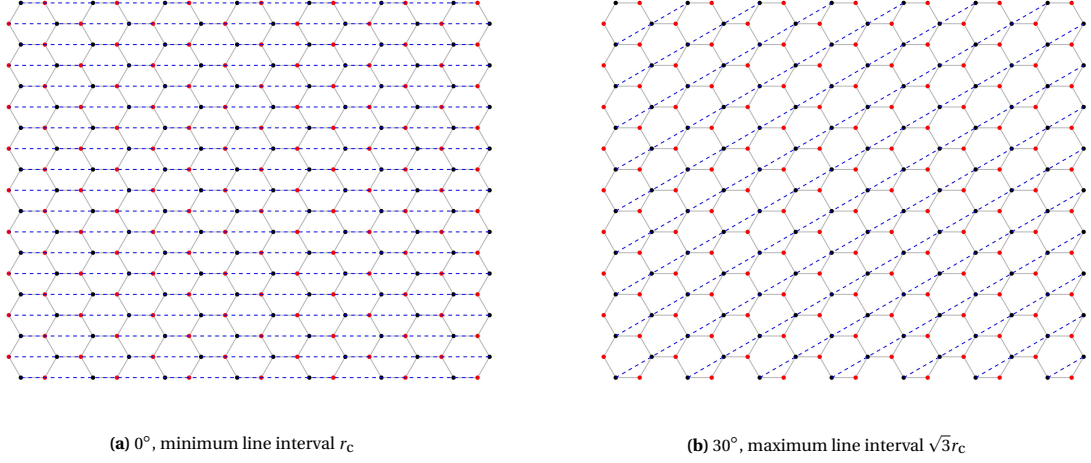
(a) 0°, minimum line interval r_c **(b)** 30°, maximum line interval $\sqrt{3}r_\text{c}$

Figure 6.10: Distinct fiber spacing along different directions

In addition, better positioning accuracy can be observed for the longer side of a rectangle, as it can be evaluated by more receiving fibers in the vicinity. Therefore, the errors from samples with different orientations are projected along the coordinate of the object referred to Equation 4.5 to obtain the errors along the object's short and long edges $\Delta x'$ and $\Delta y'$.

$$\begin{bmatrix} \Delta x' \\ \Delta y' \end{bmatrix} = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} \cos\alpha(x_\text{c} - x_0) - \sin\alpha(y_\text{c} - y_0) \\ \sin\alpha(x_\text{c} - x_0) + \cos\alpha(y_\text{c} - y_0) \end{bmatrix} \tag{6.6}$$

In the given samples, calibration boards 2 and 4 align the long edge with the dense direction, while boards 1 and 5 align the short edge. Although the most accurate sample comes from the former, measurements in the sparse direction are highly unstable. In calibration board 4, this instability even affects the results in the dense direction, with average errors of 1.15 mm and 1.34 mm along the long and short edges, respectively. The latter ensures the accuracy of the shorter edge with more stabilized overall results. In this case, the average error of $\Delta x'$ reduces to 0.38 mm, with a maximum error of less than 1 mm.

6.6. Conclusion

In this chapter, image data from the fiber array is used as input, and the planar positioning algorithm is applied to the established sensor surface prototype. A complete distributed fiber optic sensor system is constructed, and its feasibility is validated. Before measurement, the positions of each fiber in the undistorted image, as well as their maximum and minimum brightness values, are calibrated. These intrinsic parameters are stored in different tables based on the image source and used for processing the image data in measurement samples. The fibers distributed along the object's boundary are identified using the ball pivoting algorithm, and their positions, along with the fiber brightness under different illumination cases, are used as inputs for the planar positioning algorithm. Five calibration boards with a total of 20 position samples in different orientations were used to estimate positioning accuracy. Among these 20 samples, the developed distributed fiber optic sensor showed error values of 0.77 mm and 0.65 mm in planar DoFs, with an overall angular deviation of less than 1 degree. These errors mainly stem from actual brightness losses in different receiving fibers and sensitivity variations in fiber spacing at different angles within the sensing array. If the short edge of the measured object is aligned with the dense direction of the sensing array, the measurement error along the long edge can be further reduced to 0.38 mm.

7

Conclusions and recommendations

The final chapter is split into two sections, all research findings and results are summarized in the conclusions, and the recommendations are given to improve the system design and increase the potential of the sensor concept.

7.1. Conclusions

- In the semiconductor and photovoltaic industries, contactless positioning systems have been applied to handle ultra-thin wafers to reduce the risk of mechanical failure. To improve the accuracy of motion control, the positioning sensors are expected to be integrated into the systems.

- There is great potential for non-tactile positioning sensors. The extra element is required to be mounted on the measured object for electric sensing methods and the specular surface of ultra-thin wafers makes it difficult to track the displacement by optical mouse sensors.

- The distributed fiber optic sensor system is introduced. The sensor unit with a set of transmitting and receiving fibers is expanded over the measurement plane to increase the measurement range and avoid attaching additional elements to the measured object compared with concentrated fiber probes used in patent outcomes.

- Partial illumination scheme is applied to fibers distributed over hexagonal cells so that each receiving fiber is surrounded by at most one light source. This approach effectively prevents light interference from different light sources that severely affects the accuracy of previous scientific research.

- Light profiles with 2D Gaussian distributed intensity are adopted to construct the simulated fiber optic sensor array. The theoretical brightness detected by each receiving fiber is calculated by a surface integral of light intensity within the detection scope under the reflective region.

- The non-elementary integral functions in the theoretical brightness are approximated by a multivariate function dependent on edge distance, orientation, and density of fiber probes, where shifted sigmoid, trigonometric, and quadratic models are applied to these variables respectively. The function model nicely fits the theoretical results, with the coefficient of determination being no less than 0.99.

- The fitted multivariate function is validated using a prototype equipped with a single set of transmitting and receiving fibers. These fibers are mounted on a custom-developed motion control mechanism that allows precise planar positioning between the measured object and the receiving fiber. This is achieved through a sliding pair and a spline fit, offering a translation resolution of up to 0.05 mm.

- An illumination control module has been developed, capable of switching different groups of light sources on or off by programs via an Arduino microcontroller. The module also integrates the common emitter amplifier circuit so that the brightness can be manually adjusted to ensure consistency.

- The sensor output curves relating brightness to edge positions are calibrated using a range of volumetric light diffusers. PyraLed YT290 shows the greatest brightness difference between maximum and minimum values, offering a higher resolution for position decoding.

- The sensor output curves are further calibrated across different fiber densities. An 8 mm in-radius hexagonal configuration in the sensor array is chosen, achieving a resolution of 0.1 mm with a relatively sparse fiber distribution. In this setup, the measurement range of the receiving fiber is expanded to 7.2 mm and the curve's goodness of fit to the theoretical model reaches 97%.

- The obtained geometric parameters for the fiber array are applied to reconstruct the simulated fiber array, and planar positioning algorithms that determine the planar DoFs of an object using positions of receiving fibers near its boundaries, and the brightness outputs are developed based on this platform.

- The planar positioning algorithms are evaluated using 60 test samples at random positions, yielding median errors of less than 0.05 mm in both the x and y directions, and a median angular error of approximately 0.1 milli-rad. The accuracy is within 0.15 mm with a 90% confidence level, and the maximum error does not exceed 0.3 mm.

- The prototype for the distributed fiber optic sensor system has been designed and manufactured. Each light source aligns to transmitting fibers in the same row of the array by a manufactured transmitting fiber collimator, and the receiving fiber collimator gathers all the receiving fibers into the scope of the camera. Ambient light isolation and fiber positioning are two focus areas during manufacturing.

- The illumination control module for the sensor array consists of 6 LEDs for each group of transmitting fibers. Relays are adopted as digital switches for each group and an additional battery with higher voltage is used as a power supply to ensure sufficient brightness.

- The maximum and minimum brightness for each receiving fiber is required to be calibrated in advance to normalize the measured brightness as there is a great variation in the brightness differences due to discrepancies in the bending of fibers.

- The light spots of receiving fibers captured by the camera are transformed into brightness and positions on the fiber array through image processing, the data obtained by the prototype of the distributed fiber optic sensor system serves as input for the proposed planar positioning algorithms and returns estimated planar DoFs for a real reflective object.

- Among 20 samples with varying actual positions and orientations, the established distributed fiber optic sensor system demonstrates median errors of 0.77 mm in the x direction and 0.65 mm in the y direction, with an overall angular deviation of less than 1 degree.

- The measurements demonstrate good agreement with the theoretical model. The sub-millimeter accuracy differences are primarily due to brightness loss from fiber bending, which causes a shift in the measured distance between the edge and the fibers. Moreover, position errors vary with different object orientations due to the anisotropic fiber spacing in the sensor array, where the averaged accuracy can be further reduced to 0.38 mm if the short edge of the object is aligned to the dense direction of the sensor array.

- The **final conclusion** is that the distributed fiber optic sensor system with a partial illumination scheme successfully limits the overall error in planar degrees of freedom to the sub-millimeter scale, while maintaining an angular deviation of less than 1 degree. In this sensing configuration, the measurement range is extended to 210×148 mm^2, and the sensing element is no longer required to be attached to the object. Therefore, it shows a promising prospect for position measurement especially for flat and fragile products.

7.2. Recommendations

- Incorporating the gap distance between the fiber tips and the diffuser into the design parameters of the sensor surface. In practical situations, the diffusivity of the light diffusion sheet is also related to its distance from the light source. The farther the light source is from the diffuser, the stronger its diffusing capability. In theoretical models, this parameter corresponds to the standard deviation in a two-dimensional Gaussian function of a light profile, while it is set as a constant in this research. By adjusting the gap distance, the detection scope of the receiving fibers might be further expanded, thereby reducing the fiber density.

- Incorporating planar positioning methods for other regular-shaped objects into the current algorithm. The distributed fiber optic sensor system developed is currently limited to detecting rectangular objects, while fibers near the boundaries of the object can be extracted. By judging the shape of the object

through the positional distribution pattern of these fibers, other methods used to detect circular or square objects can be integrated into this system.

- Try to reduce the bending of transmitting fibers during fiber installation. Due to the inconsistent distance from the light source to the position of each transmitting fiber in the sensor surface, the transmitting fiber is bent to varying extents. This causes a great variation in brightness detected by receiving fibers and affects the estimations of the distance between the fiber and the boundaries. In the new layout, it is possible to replace the LED lamps with light strips, so that the light sources can be placed directly below the positions of the transmitting fibers on the sensor surface.

- Improving the configuration of position calibration devices. In the prototype of the distributed fiber optic sensor system, each real position corresponds to a 3D-printed position calibration board. The production of these boards wastes time and consumes a lot of materials, resulting in a limited number of measurement samples.

- Integrating light-switching and snapshot programs into the position estimation process. These are performed manually step by step in this research for measuring a static object. For dynamic motion tracking, these processes should be performed automatically in sequential order, and the latest three snapshots are used to derive the current position. Other issues should also be taken into consideration to realize real-time measurements, such as the maximum shining frequency of light sources that can be captured by the camera, and the run time limitation of positioning algorithms.

<div style="text-align: right; font-size: 3em;">A</div>

Literature Review and Project Proposal

A.1. Introduction

A.1.1. Motivation

In the manufacture of small-sized, high-performance devices like solar cells, power chips, and other semiconductor products, the ultra-thin wafer finds extensive use. The market study indicates that due to the link between miniaturization and improved performance, the demand for thinner wafers below 200 μm has been increasing recently, with a compound annual growth rate (CAGR) of more than 5% in the total market volume.
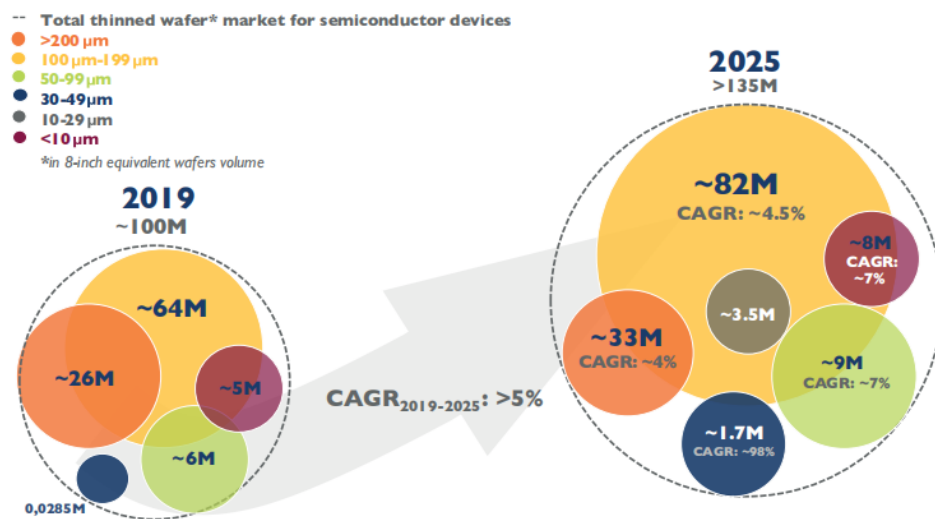
Figure A.1: Thinned wafer market volume: 2019-2025 breakdown by thickness range[2]

The market share and predicted trend of different consumer products made by thinned wafers is illustrated in Figure A.2. The overall thinning equipment market was worth almost 461 million dollars in 2019 and will exceed almost 792 million dollars by 2025 mainly generated by memory, CMOS image sensors, and power components. Although there are similar trends for thinnings applicability in semiconductor applications, the reasons for using such techniques differ from one device to another and depend on the end applications.
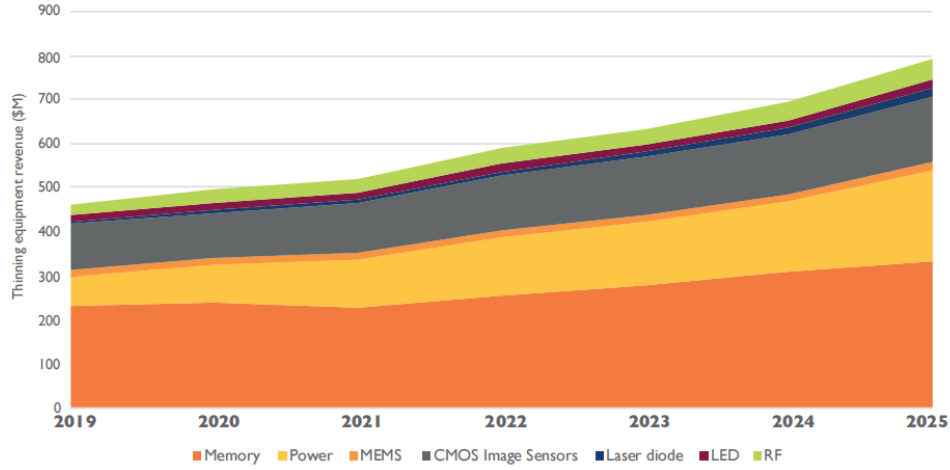
Figure A.2: Thinning equipment market per device: 2019 vs 2025 forecast[2]

The power device is a semiconductor that functions as a switch or rectifier in power electronics. It has a wide range of power electronics applications in smart appliances, including electric vehicles, wind power systems, and solar photovoltaic panels. Power MOSFETs account for the vast majority of the power transistor market. Si-based MOSFETs use wafers with thicknesses ranging from 50 μm to 55 μm on 300 mm diameters. Thin wafers are required here because reduced thickness reduces on-resistance, increases current carrying capability, and reduces power consumption. Image sensors are typically made up of a stack of sensor chips, cover glass, and wire bonds connected to an application-specific integrated circuit. All three wafers are thinned not just to lower the device's size, but also to increase the integration of process electronics into the pixels. Sensor wafers with severe thinning below 100 μm have more sensitive and complex architectures with high connection density, resulting in greater light sensitivity.

(a) Power device[43]

(b) CMOS image sensor[10]
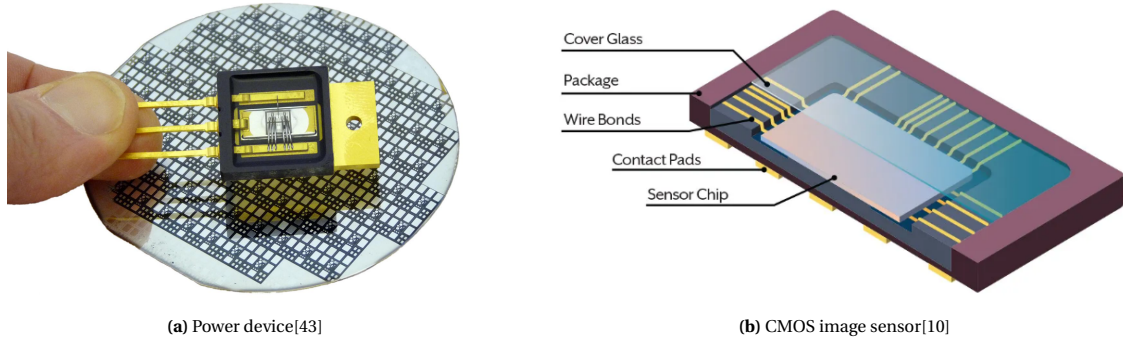
Figure A.3: Application of thin wafers in consumer products

A.1.2. Problem statement

Wafers having a thickness of less than 100 μm are mechanically flexible and difficult to handle directly by end effectors, increasing the risk of mechanical failure and contamination. Existing research has provided information on contactless actuation systems for the control of motion of thin and flat substrates on the operational platform using various tactics such as aerostatic or magnetic levitation[52, 65]. An extra sensorial system can achieve greater operating platform performance and higher motion accuracy of the handled object than proposed actuator solutions with sensor-less manipulation. The primary elements of the position measuring system are summarized below about the characteristics of the operational platform and the measured object.

- The sensor only measures in-plane degrees of freedom($XY\theta_z$) under out-of-plane constraints;
- The measured object is thin and flat, with an exact 2D geometric shape;

- A large motion stroke in macro size is required for transmitting the measured object;

- Tactile sensing techniques and the connection of sensing elements to the measured object are not expected;

- The physical, chemical, or other attributes of the measured object cannot be changed to accommodate the sensing system.

Though position sensors for measuring multiple dimensions have been extensively investigated and developed, and the majority of them are now accessible on the market, they continue to confront obstacles in meeting all of the requirements of precision positioning systems. The simplest multiple-axis measurement system consists of independent 1D position sensors, with measurement data in different dimensions layered to estimate multi-DoF displacement. This measuring scheme is ideally suited to linear stages with low cost and good accuracy, whereas the detectable angular deflection of the measured item in big stroke leads to a substantial Abbe error, where the movement cannot be simply detached into linear displacement in different dimensions[14, 15].
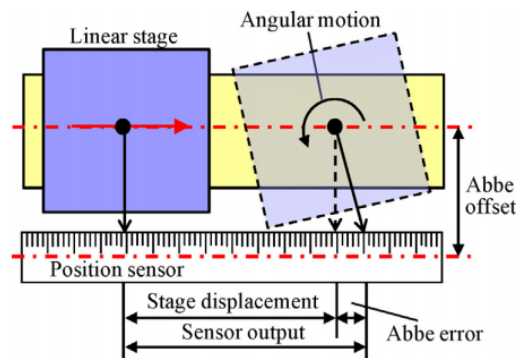
Figure A.4: Abbe error[14]

The Abbe error restriction in multi-axis measurement highlights the need for decoupling strategies between multi-DoFs data gathered by position sensors, with the Charge Coupled Device (CCD) camera being one of the common options. The CCD camera is mounted over the operational platform that covers the object's whole motion range in its visual scope. This measurement scheme does not necessitate any complex physical installations on the platform. However, the increased visual field caused by significant motion strokes reduces measurement resolution[16], and several cameras covering the operational platform might be difficult to implement in a small space and costly[16, 25]. Furthermore, other obstructions, such as manufacturing devices over the handled object, may block the view of the CCD camera from above, resulting in interference in position measurement.
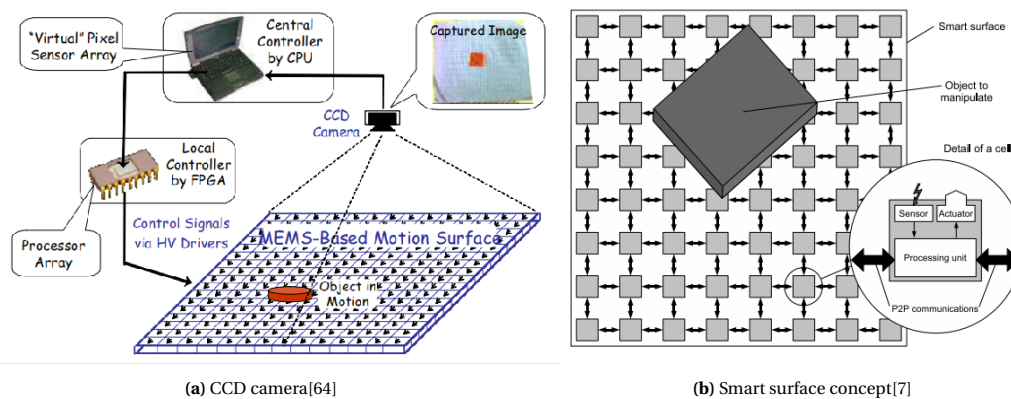

(a) CCD camera[64]

(b) Smart surface concept[7]

Figure A.5: Configuration of multi-axis measurement methods

In contrast, the smart surface concept, which attempts to add intelligence to the actuator array in the operational platform by integrating proximity sensors into each actuator cell, is a promising way to achieve precise object positioning. In addition to overcoming limitations in previous sensing methods, this bottom-viewed sensing approach offers new possibilities in the application with high robustness to failures due to the redundancy of mechanisms[7], which may be an inspiration for the following research.

A.2. State-of-the-art research

A.2.1. Search strategies

This section illustrates a systematic technique for extracting a list of referable research from vast literature reviews to gain insights into the state, trend, and underlying concerns of planar positioning sensors in precision positioning systems. The papers were acquired from Google Scholar, Google Patent, and the TU Delft repository.

Table A.1: Iteration over keywords in literature research

	Initial attempt	Optimized trial
Sensor features	"distributed sensor" **OR** "smart surface" **OR** "sensor array" **OR** "sensor surface"	"distributed sensor" **OR** "smart surface" **OR** "sensor array"
Objective	"displacement measurement" **OR** "edge detection" **OR** "object detection"	"displacement measurement" **OR** "position measurement"
Function	"planar" **OR** "2D"	"planar" **OR** "2D" **OR** "in-plane"
Constraint	contactless	levitated

To begin, a collection of keywords and their paraphrases was constructed by extracting potential features of precise positioning systems. Throughout several trials and improvements, a higher proportion of referable papers can be obtained by the keyword combinations in the right column in Table A.1. The second phase involved extracting common properties among useful results as new search terms. The strings are condensed like "2D position measurement" or "planar position measurement" to discover more sensing mechanisms. The search results were also broadened by extracting additional articles from reference lists with highly relevant ideas in the title and abstract.

In addition to the literature survey, an extra patent search was carried out over the promising approach to explore any new practical ideas in the design of sensing systems. The most relevant descriptions of the Cooperative Patent Classification (CPC) can be identified and used to guide more useful patents.

Table A.2: Descriptions of CPCs from relevant patents discovered by exact keywords

Code	Ref.	Description
G01D5/268	[27]	Mechanical means for transferring the output of a sensing member; Means for converting the output of a sensing member to another variable where the form or nature of the sensing member does not constrain the means for converting; Transducers not specially adapted for a specific variable characterized by optical transfer means, i.e. using infra-red, visible, or ultra-violet light using optical fibers
G01D5/34723	[27]	Scale reading or illumination devices involving light-guides
G03F7/70775	[17]	Position control, e.g. interferometers or encoders for determining the stage position
G03F7/7085	[17]	Detection arrangement, e.g. detectors of apparatus alignment possibly mounted on wafers, exposure dose, photo-cleaning flux, stray light, thermal load
H02L21/68714	[17]	Apparatus specially adapted for handling semiconductor or electric solid state devices during the manufacture or treatment thereof

From limited patents discovered by exact keywords, G01D5/268 provides a comprehensive summary of the working principle of the specified sensing system. Based on the main idea of the abstract of patents obtained by the CPC code, 11 referable investigations were selected from the top 25% of 579 results, followed by consecutive irrelevant patents in over 10%.

A.2.2. Sensing schemes overview

Overall, 39 articles have been discovered based on a literature review of planar positioning sensors. Appendix A.5 extracts and summarizes the performance and application fields of each sensing system, and these search results will be utilized to support the state-of-the-art studies in the following sections. The statistical graphs

show a rising trend in research on planar positioning sensors, with six major types of planar position measuring systems presented. Optical approaches predominate over electric methods, with more interest in optical encoders and photodiodes compared to evenly distributed research on capacitive and magnetic sensors. This distribution pattern provides an implicit perspective of sensor performance, with electric methods revealing more underlying possibilities and optical methods perhaps more applicable to present research. The working principle of position sensors in various categories will be discussed further to identify any potential gaps in their real-world implementations on contactless transport systems.
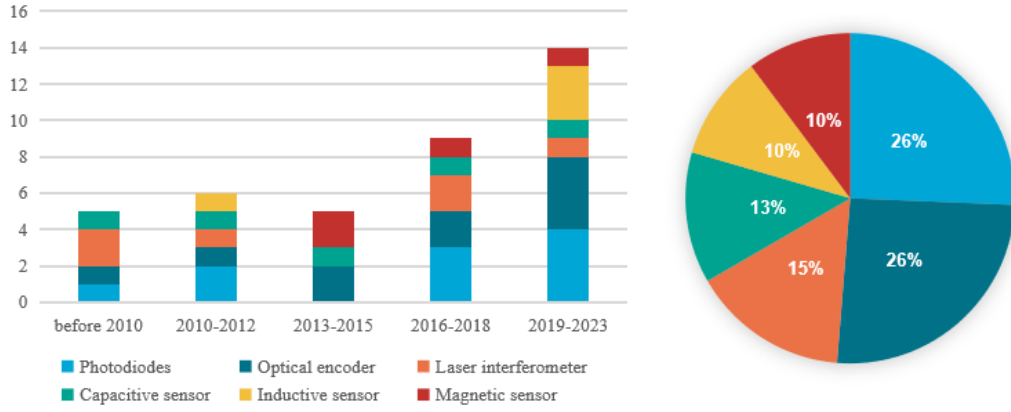
Figure A.6: Development of various sensing systems throughout time and their overall proportions

Electric methods

Electric 2D position measurement systems are distinguished by their inexpensive cost and compact size. By incorporating the smart surface concept with distributed sensing elements into the measuring system design, the resolution can be increased to micron scales. Figure A.7 shows representative electric methods for sensing in-plane motions of planar objects. An array of active cells on the fixed plate generates an exact electric or magnetic field over the surface, which can be stated directly using in-plane DoFs. Different decoding algorithms, depending on the characteristics of the sensing medium and sensor arrangement, are used to calculate the planar position of the measured object via the output of the sensor array.
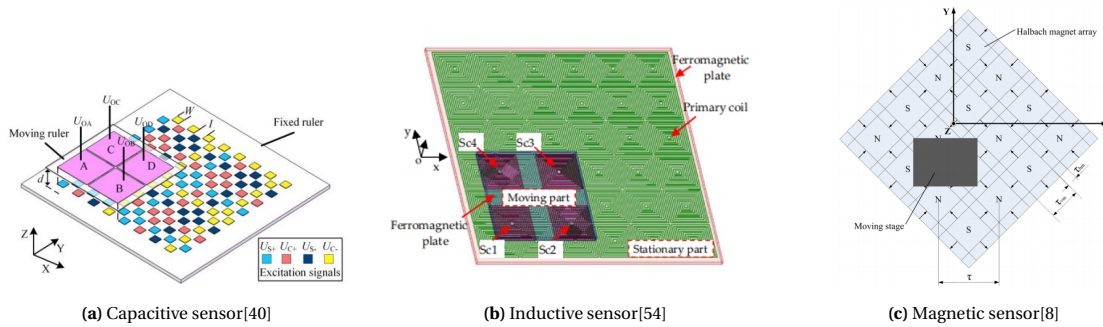

(a) Capacitive sensor[40] (b) Inductive sensor[54] (c) Magnetic sensor[8]

Figure A.7: Electric sensing methods

Peng et al. used the periodic variation pattern in capacitance to measure 2D displacement on a large scale. The overlapping area of capacitive electrodes is linearly dependent on relative motion between receiver electrodes and the array of excitation electrodes on the fixed plate over each period[40]. Wu et al. proposed an inductive sensing scheme that uses the Fast Fourier Transform (FFT) to obtain the amplitude and initial phases of induced voltages on moving objects. Rotation iteration algorithms were used to decouple the trigonometric relationships between the relative motion of the secondary coils and the voltage signals and measure the in-plane displacement[54]. Chen et al. used a uniaxial Hall sensor array with a highly symmetrical architecture to map magnetic detection to location. Each Hall sensor produced a voltage output proportional to

flux density normal to the sensor surface, while flux density from the Halbach magnet array is a multivariate sinusoidal function in planar coordinates. The approach used multi-dimensional nonlinear equations to decouple planar position and orientation from flux densities, resulting in univariate equations and estimations of each planar DoF by solving 1D optimization problems[8].

Optical methods
The optical methods inter planar motion or position by monitoring changes in light intensity, phase, or wavelength. Figure A.8a depicts a laser interferometer using a Polarizing Beam Splitter (PBS) that divides a dual-frequency laser into two distinct light pathways. The laser component transmitted through PBS incident on the mirror plane installed on the edges of the moving target and reflects with a frequency shift due to the Doppler effect[14]. This frequency shift can be obtained by comparing the frequency difference of different laser components to estimate the instantaneous displacement of a moving target. Because of its nano-scale resolution over centimeter-level motions, this sensing technique is commonly used in high-precision linear stages with long motion strokes.

(a) Laser interferometer
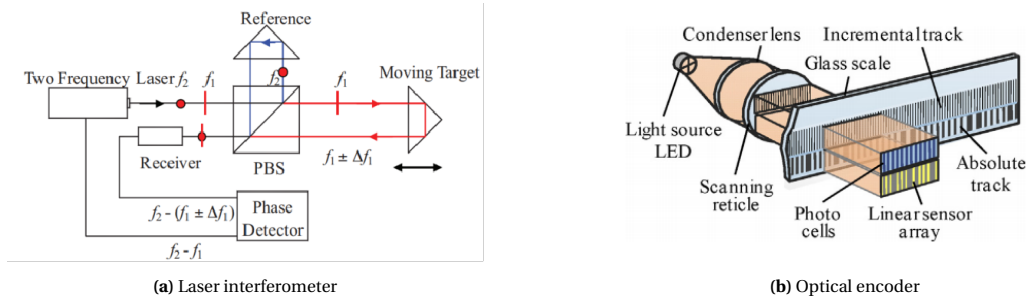
(b) Optical encoder

Figure A.8: Optical sensing methods[14]

The optical encoder typically consists of a scale grating with incremental and absolute tracks, as seen in Figure A.8b. When light is projected on the optical head through the absolute grating tracks, a special light pattern can be recognized due to the unique serial code structure over the measuring range of the encoder, which provides information on the absolute positions[14]. The fine and homogeneous incremental grating is used to quantify relative displacement, which improves positioning precision. Kim et al. used phase-encoded binary scale to planar encoders by superimposing single-track code structures orthogonally to measure the 2D absolute planar position[31]. This research was later improved by the color-encoding method, where the detector can distinguish multiple color channels over bar codes[30]. When bar codes are created at very tiny scales, it is possible to identify entire rotation angles of the measured object with sub-micron accuracy.
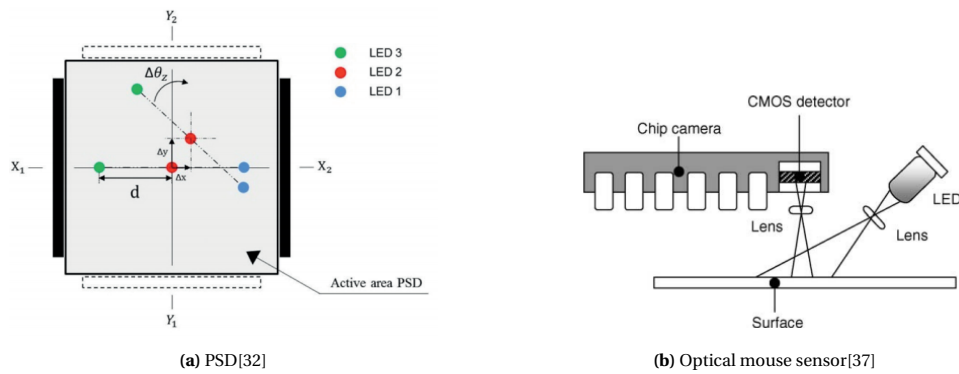

(a) PSD[32]

(b) Optical mouse sensor[37]

Figure A.9: Application of photodiodes in planar motion measurement systems

A photodiode is a PN junction device that generates current when exposed to light. This photoelectric element has been integrated into position sensors by various methods. For each configuration, the object is illuminated by specific light sources such as Light Emitting Diodes (LED), and an array of photodiodes is used to collect the intensities of reflected light from the object, with the intensity pattern determined by the

location of the measured object. Lampaert et al. estimated the in-plane DoFs by moving three light spots relative to the photosensitive area of a Position Sensitive Detector (PSD), where the position of the light spots can be calculated using photo currents acquired by two pairs of electrodes in the X and Y directions[32]. Figure A.9b shows a typical optical mouse sensor arrangement with a light source and detecting unit positioned on the operational platform[37]. Because the diffusive reflection pattern is distinct throughout the rough surface, the in-plane displacement and rotation can be determined by extracting the motion of identical local intensity patterns in consecutive frames from the CMOS detector.

Fiber optic sensors use a distributed array of transceiver fiber bundles to detect the object's location. The optical fiber consists of a core and cladding that have distinct reflecting indexes. Because of this difference, the light beam travels through the core by repeatedly bouncing off the cladding wall with little loss in the light intensity and is disseminated at the other end at a specific angle. Each transmitting fiber sends out a Gaussian profile of light spots over the plane while receiving fibers are used to measure the reflected light intensities at their respective positions.

(a) Light transmission mode[38]

(b) Sensing configuration[13]
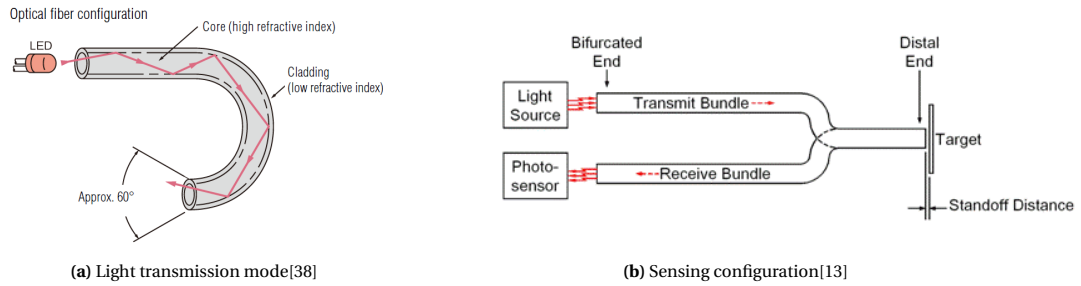
Figure A.10: Fiber optic sensor

Among the discovered patents, Joseph et al. used light pattern change dependent on the gap distance between fiber tips and object surface to measure the spatial motion of the target plane. As illustrated in Figure A.11a, the reflected light from the target surface(210) formed a ring-shaped pattern on the corresponding imaging sensor in the sensing array(230), and the size depends on the gap distance[28]. When the target surface moved relative to the optical path array, imaging sensors captured the shift of ring-shaped patterns to evaluate displacement in X and Y directions.

(a) Gap distance variation[28]
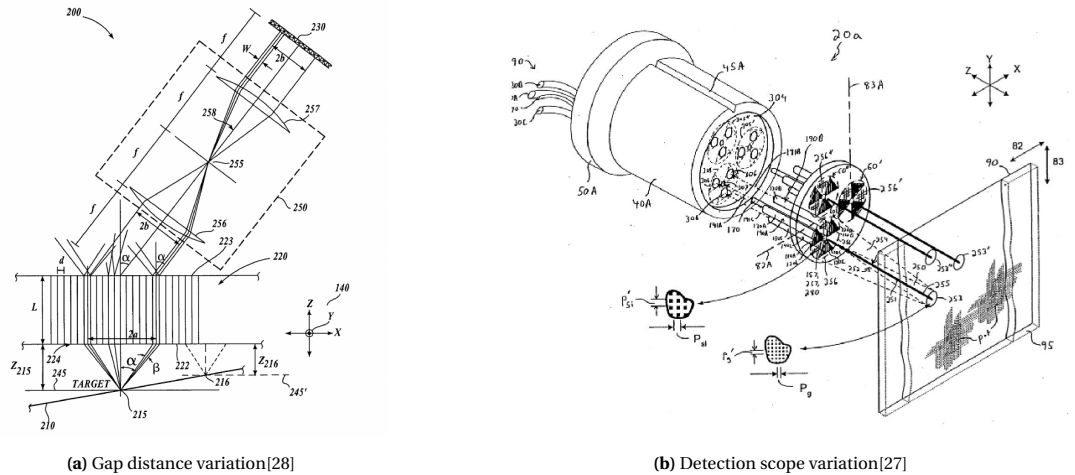
(b) Detection scope variation[27]

Figure A.11: In-plane positioning patents by fiber optic sensors with distinct measurement principles
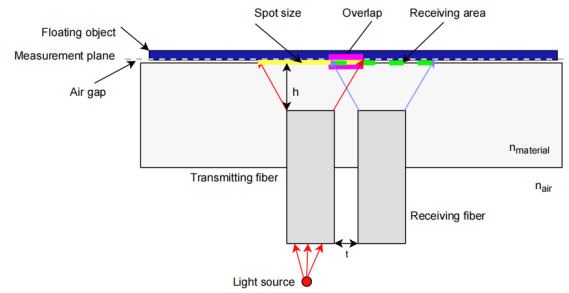
For strictly controlled out-of-plane motion with a set fly height, grating scales can be employed to regulate the overlapping region between light spots and the visual scope of receiving fibers. In Figure A.11b, three sets of receiving fibers are covered by grating masks(256, 256', 256") in horizontal or vertical directions, with various grating scales along X and Y directions on the moving plane[27]. When the fiber probe(40A) moves

relative to a grated planar target(95), the periodic intensity fluctuations received by different groups of fibers can be used to determine in-plane displacement.

Unlike detector heads with packed fibers in the previous patent, fiber transceiver sets can be spread over the measuring plane to expand the measurement scope without modifying the handling item. According to the prototype built by bachelor students at TU Delft in 2019, the photo-sensor beneath the measurement plane determines if the object is placed above the receiving fibers based on the brightness of the light spot from corresponding fiber tips[12]. This sensing setup could span a wide detection range, but a millimeter scale accuracy was obtained due to detection in binary intensity states. To take full advantage of light intensity levels, this distributed fiber optic sensing system was improved in subsequent research by adding a light diffusing layer between the optical fibers and the measured object. The diffuser expanded the light spots from transmitting fibers and the detection scope of receiving fibers, allowing for a smooth transition pattern of light intensity to estimate the distance between the receiving fibers and the edges of measured object[4, 36].
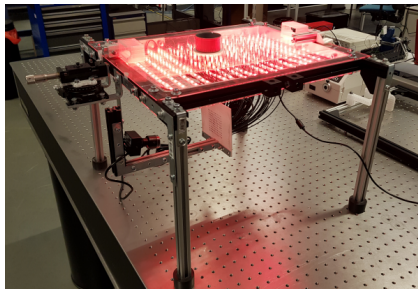
(a) Prototype setup[12]   **(b)** Light expansion by diffuser[4]

Figure A.12: Smart surface concept by fiber optic sensors

A.2.3. Research gap

Many undesired aspects may limit the use of planar positioning sensors in a contactless wafer transfer system. Inductive sensors cannot be used for non-metals where the eddy current can not be induced. Laser interferometers exploit the frequency shift of the laser beam reflected from the sides of the moving target to estimate the instantaneous displacement, while it is not applicable for measuring ultra-thin objects. In the case of optical encoders, a mechanical defect on local features distorts the shape of the entire grating, resulting in full failure due to the high susceptibility of complicated planar grating scales, whereas the Abbe error exists in the stack of linear encoders. Table A.3 lists the performance of remaining sensing systems, providing an explicit and systematic assessment of their desired features and existing constraints from multiple evaluation criteria.

Table A.3: Performance overview of different sensing schemes

	Electric method		Optical method			
	Capacitive [6, 24, 40, 58, 59]	Magnetic [8, 16, 35, 62]	PSD [22, 25, 32]	Optical mouse [1, 37]	Fiber optic sensor	
					Concentrated [27, 29]	Distributed [4, 12, 26, 36]
Sensor attachment	electrodes	Hall sensors	LED holder	-	gratings	-
Object features	conductive	ferromagnetic	-	coarse surface	-	specular surface
Achievable range	<200 mm	<50 mm	<10 mm	<2 mm/step	<15 mm	≈200 mm
Accuracy	micron	sub-millimeter	micron	micron	micron	millimeter
Cost	cheap	cheap	fair	cheap	expensive	cheap
Robustness	fragile	pass	excellence	pass	excellence	pass
Manufacture	easy	pass	easy	easy	pass	hard

From the perspective of the operating principles of planar positioning sensors, all sensing configurations, except optical mouse and distributed fiber optic sensors, attach additional parts to moving objects, which is undesirable, particularly for contactless handling systems. Compared to displacement measurement by

optical mouse sensor, the smart surface concept can be used in distributed fiber optic sensors to determine absolute position over a wider range. The specular surface properties of the detected object also make it more favorable for the application of this sensing system compared to other alternatives.

Nevertheless, this sensing scheme still has significant space for improvement in other aspects, such as accuracy. According to recent research literature, these limited accuracies are primarily attributed to simple detection algorithms along 1 dimension or deficiencies in the arrangement of physical platforms such as the millimeter-scale clearances in the assembly of the prototype and poor resolution for real object positions. These defects can also be indicated by poor stability and small intensity differences between steady intensity levels in the intensity curves from the experimental results. Therefore, the potential of the distributed fiber optic sensors in planar positioning will be fully exploited in this research.
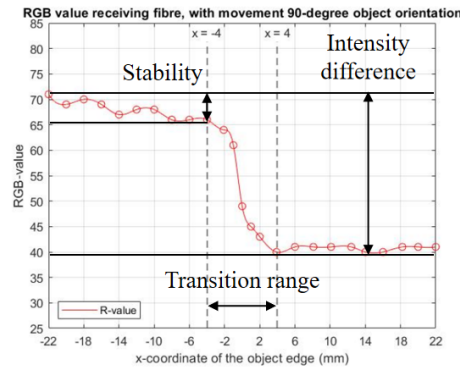
Figure A.13: Relationship between transitional light intensity and edge locations[36]

A.3. Research objective

The related primary research question with many sub-questions is proposed below.

- How can the transitional light patterns produced by fiber optic sensors be used to accurately position a flat and specular object?
  - What factors other than fiber coverage influence the precision of the measured sensing medium?
  - How can the distributed fiber optic sensors effectively decouple the planar degree-of-freedom of the measured object through fiber positions, intensities, and other relevant parameters?
  - What is the minimum required fiber density to achieve sub-millimeter scale accuracy in the positioning of the object?
  - In what ways does sufficient ambient illumination impact the planar positioning of the object?

Accordingly, the research objective is stated as:

"Theoretical and experimental determination of the precision and accuracy of a distributed fiber optic sensor system intended to determine the planar position of thin substrates, and derivation of optimal design parameters for this sensor system."

A.4. Research plan

The flow chart of the scheduled research progress is shown below, and it mainly consists of 3 steps: sensor setup in a unit cell, construction of the smart surface, and further improvement on the positioning method considering light interference. The planar positioning methods will be developed on the simulation platform and verified by the relevant experiment, with specific tasks listed in the calendar schedule in Appendix A.6.

The adjustable parameters in sensor design, such as detecting scope, light profiles, and fiber distance, that affect the sensing medium, should be mathematically modeled in the first step of simulation to provide an initial estimation model of light intensity over fiber coverage. Control parameters other than fiber coverage that may affect sensor output should be identified through experiments on a single set of fiber optic sensors. Improvements to the prototype's electronic, mechanical, and optical components will be made to approach the ideal intensity curve with consistent RGB values at steady states, a sufficient transition range, and a clear

intensity difference between steady levels.

With a determined mathematical model, an algorithm that extracts the measured object's planar degrees of freedom from the simulated intensity outputs as well as the corresponding fiber positions is expected to be developed. The precision of simulation can identify the size of design parameters such as fiber density, resulting in reliable basic parameters for the design of the smart surface with numerous sets of fiber optic sensors in the next experimental step.

The smart surface is also planned to accomplish position measurement of the planar object under sufficient illumination conditions, with a fluorescent lamp widely used in laboratories located above. The receiving fiber along the object edge has a transitional color between white and red, which distinguishes it from totally covered or uncovered instances. The color indicates the distance between the receiving fiber and the edges, whilst the position of fiber sets in transitional light patterns can be utilized to infer the orientation of a planar object. An extra CCD camera will be mounted atop the smart surface to calibrate the sensor's performance, including accuracy, resolution, and robustness.
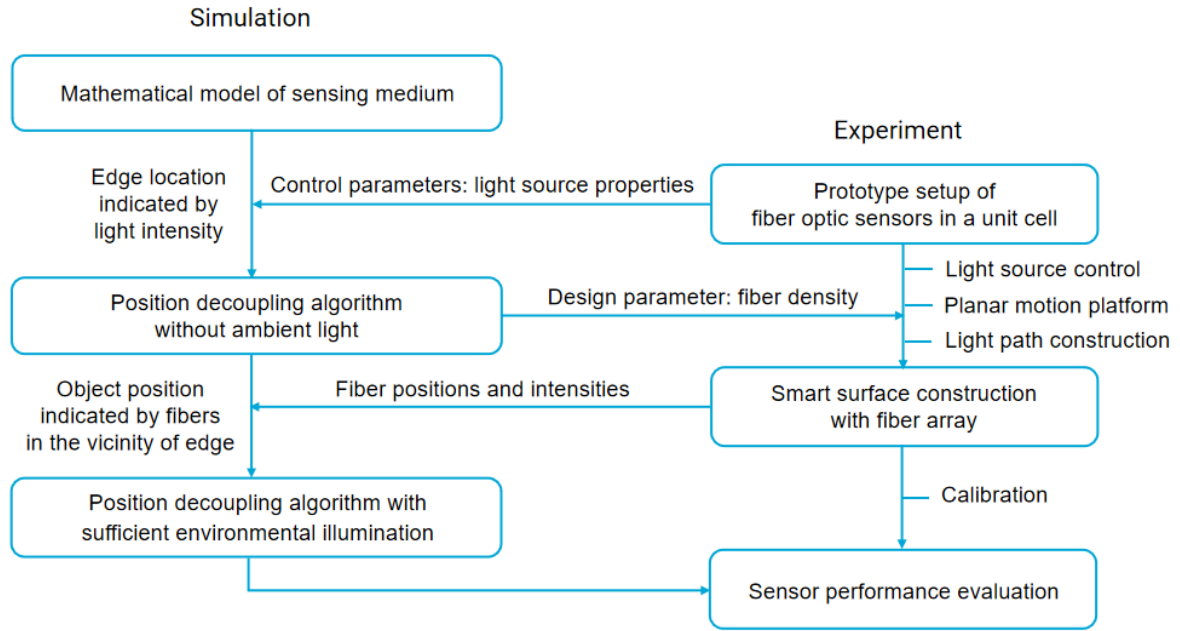
Figure A.14: Overall research plan

Preliminary tests on initial simulation and experimental steps have been performed to prove the feasibility of the approach, with the first results introduced in the following subsections.

A.4.1. Mathematical modelling of sensing network

(a) Illumination case 1   **(b)** Illumination case 2   **(c)** Illumination case 3
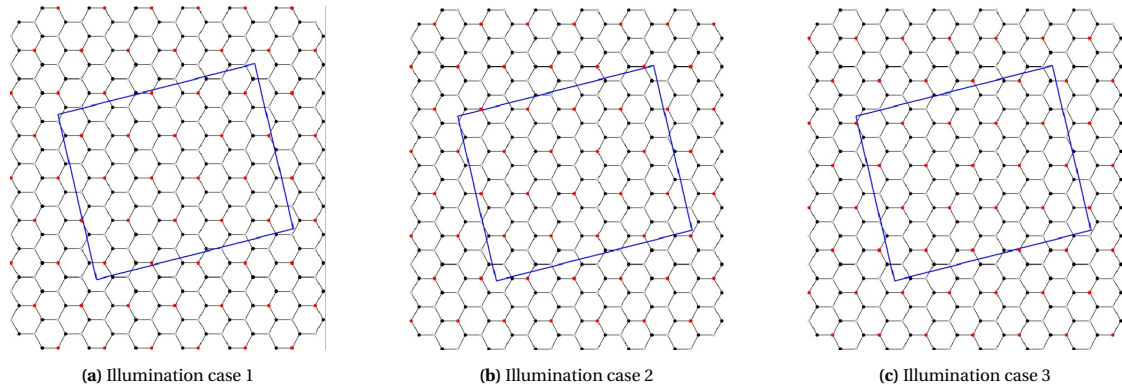
Figure A.15: Partial illumination by different groups of transmitting fibers in sensing surface

Figure A.15 illustrates the MATLAB simulation framework for the sensing surface. The transmitting and receiving fiber tips, shown by red and black dots, are organized in a hexagonal array, while the flat object, represented by the blue rectangle, can be freely positioned in the plane. Transmitting fibers are divided into three groups and sequentially illuminated to ensure that each receiving fiber is surrounded by at most one light source. This sensor arrangement helps mitigate the effects on measurement results caused by interference of light from different positions.

Existing literature research[26, 61] show that light power emitted from plastic optical fibers is Gaussian distributed over the measurement plane. Given a certain gap distance, the empirical formula for light power can be stated as

$$P(x, y) = \begin{cases} C_0 \exp(-\dfrac{(x - x_t)^2 + (y - y_t)^2}{2(r_0\sigma)^2}) & (x, y) \in S_r \\ 0 & (x, y) \notin S_r \end{cases} \tag{A.1}$$

where C_0 is a constant for a stable light source and (x_t, y_t) is the center of the light profile emitted by the transmitting fiber with an effective radius of r_0. In the ideal light pattern, the detection scope of the receiving fiber is equal to the size of the light spot and it can be defined by the set $S_r = \{(x, y) \in \mathbb{R}^2 \mid (x - x_r)^2 + (y - y_r)^2 \le r_0^2\}$, where (x_r, y_r) denotes the position of the receiving fiber. σ is the standard deviation of the Gaussian function. If the effective radius of a single light spot is defined such that the light power on the boundary of the spot is 1% of the maximum light power at the center, the standard deviation can be calculated as

$$C_0 \exp(-\frac{r_0^2}{2(r_0\sigma)^2}) = 0.01 C_0 \exp(-\frac{(x_t - x_t)^2 + (y_t - y_t)^2}{2(r_0\sigma)^2}) \tag{A.2}$$

$$\sigma = \frac{1}{\sqrt{-2\ln 0.01}} \approx 0.33 \tag{A.3}$$

There is a high degree of similarity between light distribution profiles obtained by experimental results and the mathematical model, while the model extends the distribution to two-dimensional space to effectively estimate the planar degree of freedom.
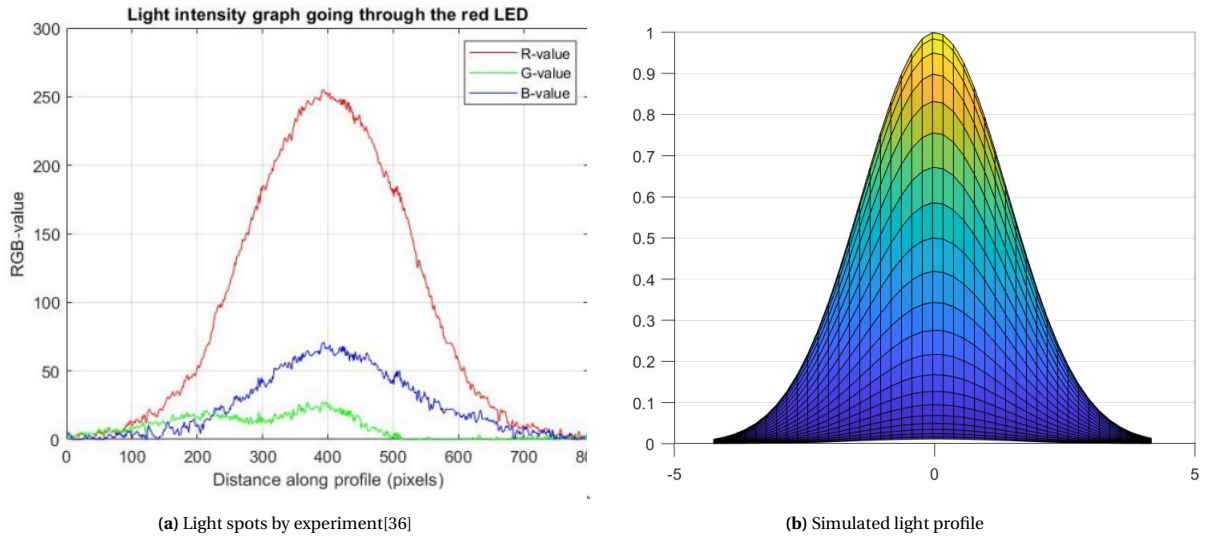

(a) Light spots by experiment[36]

(b) Simulated light profile

Figure A.16: Comparison of light profiles

For a given rectangular object with width w and length h, the mathematical expression of the enclosed region S_o dependent on the center position and orientation is derived as follows. The explicit equation of the symmetric axis l_x across the center (x_0, y_0) with an angle θ can be written as

$$y - y_0 = \tan\theta(x - x_0) = \frac{\sin\theta}{\cos\theta}(x - x_0) \tag{A.4}$$

The other symmetric axis l_y, which is orthogonal to l_x as illustrated in Figure A.17, can also be stated explicitly as

$$y - y_0 = \tan(\theta + \frac{\pi}{2})(x - x_0) = -\frac{\cos\theta}{\sin\theta}(x - x_0) \tag{A.5}$$
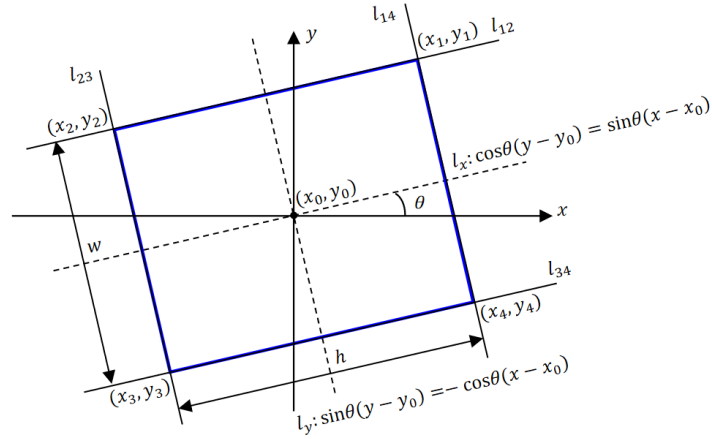
Figure A.17: Mathematical parameters of measured object

The object's edges in the length direction denoted as l_{12} and l_{34} in Figure A.17, are at a distance of $w/2$ from the symmetric axis l_x. Using these geometric relations, the associated edge functions are written as

$$l_{12} : y - y_0 = \frac{\sin\theta}{\cos\theta}(x - x_0) + \frac{w}{2\cos\theta} \tag{A.6}$$

$$l_{34} : y - y_0 = \frac{\sin\theta}{\cos\theta}(x - x_0) - \frac{w}{2\cos\theta} \tag{A.7}$$

To avoid the case of infinite slopes, the parameters in denominators are eliminated by a product of $\cos\theta$ on both sides of the equations above to yield the implicit form shown below.

$$l_{12} : \cos\theta(y - y_0) - \sin\theta(x - x_0) - \frac{w}{2} = 0 \tag{A.8}$$

$$l_{34} : \cos\theta(y - y_0) - \sin\theta(x - x_0) + \frac{w}{2} = 0 \tag{A.9}$$

It should be noted that the object center (x_0, y_0) must be bound by the parallel edges above. In this case, these implicit equations are changed into inequalities in which the coordinate values are put into the equations to determine their signs, with $l_{12}(x_0, y_0) = -\frac{w}{2} \le 0, l_{34}(x_0, y_0) = \frac{w}{2} \ge 0$. The region S_o can be described using the inequalities stated below, after repeating the process for the other pair of edges.

$$
\begin{aligned}
S_o = \{(x, y) \in \mathbb{R}^2 \mid \cos\theta(y - y_0) - \sin\theta(x - x_0) - \frac{w}{2} &\le 0 \\
-\cos\theta(y - y_0) + \sin\theta(x - x_0) - \frac{w}{2} &\le 0 \\
-\sin\theta(y - y_0) - \cos\theta(x - x_0) - \frac{h}{2} &\le 0 \\
\sin\theta(y - y_0) + \cos\theta(x - x_0) - \frac{h}{2} &\le 0\}
\end{aligned}
\tag{A.10}
$$

The previously identified inequalities also show the area in which light can be reflected. The effective region where light can be observed by the receiving fiber is $S(x_0, y_0, \theta) = S_o \cap S_r$ when the detection scope S_r is partially covered by the object, as seen in the blue region in Figure A.18. As a result, the relevant light intensity can be determined using a surface integral of light power over the effective region.

$$I(x_0, y_0, \theta) = \iint\limits_S P(x, y) \, dx \, dy = \int\limits_{x_r - r_0}^{x_r + r_0} \int\limits_{y_r - r_0}^{y_r + r_0} P(x, y) \delta(x, y, x_0, y_0, \theta) \, dx \, dy \tag{A.11}$$

$$\delta(x, y, x_0, y_0, \theta) = \begin{cases} 1 & (x, y) \in S \\ 0 & (x, y) \notin S \end{cases} \tag{A.12}$$

In the equations above, a delta function for object positions is used to convert the complicated arc boundaries into rectangular bounds, with the light power outside of the effective light reflection zone set to zero.
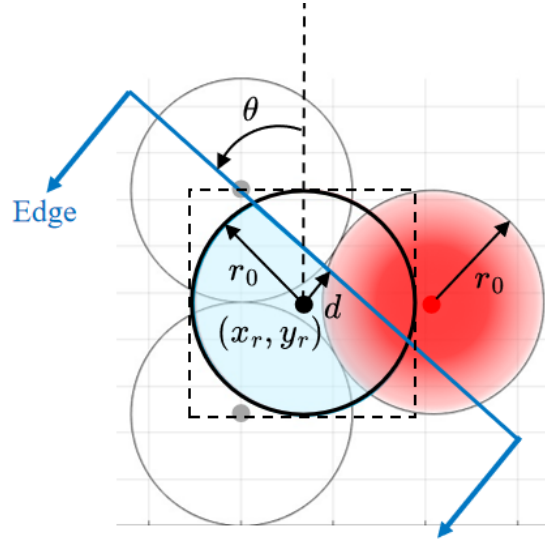

Figure A.18: Transitional intensity pattern when $S_r \not\subset S_o, S_r \cap S_o \neq \varnothing$

When an object completely obscures the detection scope, the maximum light intensity with a constant value I_m returns with $S = S_r$. If the detection scope is uncovered, $I = 0$ and $S = \varnothing$. The determination of the intensity range enables normalization over light intensity, resulting in a relative value between 0 and 1.

$$n = \frac{I(x_0, y_0, \theta)}{I_m} \in [0, 1] \tag{A.13}$$

After canceling the unknown constant C_0 by normalizing the light intensity, the remaining integrals can be calculated by MATLAB to imitate the receiving fibers' sensing process. The relationship between relative intensity n, the distance between edge and receiving fiber d, and edge orientation θ is illustrated in Figure A.19. Though one measured intensity correlates to distinct measurement results, the orientation can be firstly estimated by the positions of receiving fibers in the sensor array with transitional relative light intensities between dash lines.
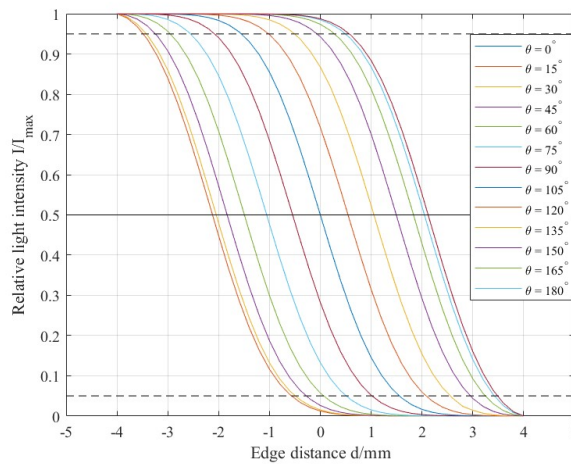

Figure A.19: Relationship between relative intensity and in-plane DoFs

A.4.2. Prototype setup of fiber optic sensors

Apart from the coverage of the object over the optical fibers, the measurement results, in reality, were also affected by other physical interference determined by experimental tests on the previous prototype as tabulated in Table A.4. These factors should be taken into account in the prototype setup.

Table A.4: primary sources of interference on sensor performance

	Control parameters				Design parameter
	Linear motion mechanism	Clearance	Surface quality	Diffuser type	Fiber density
Transition range	✓			✓	✓
Stability		✓	✓		
Intensity difference				✓	

(a) Assembly diagram   (b) Cross section view
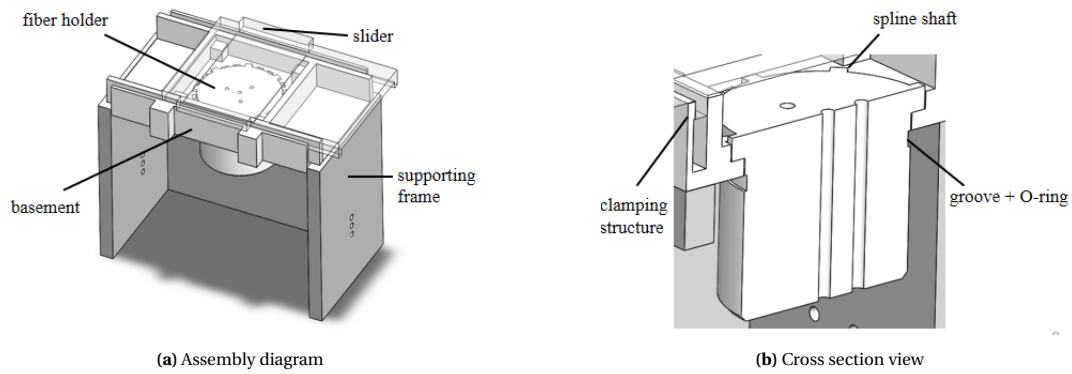
Figure A.20: Preliminary modeling on planar motion mechanism

From the perspective of mechanical design, a relative planar motion mechanism between sensing objects and optical fibers has been developed in this plan. As illustrated in Figure A.20, the specular surface is attached to the bottom surface of the slider, which can be translated along the grooves of the basement, while motion in other directions is limited by a clamping structure. The spline shaft structure on the fiber holder allows for the positioning of optical fibers in several orientations, while the axial positioning of the holder is achieved by an elastic O-ring and a groove. The extended blocks on the slider and basement enable calibrations by caliper with a resolution of 0.05 mm. The clearances between the fitting parts in the assembly diagram are limited to 0.1 mm for tight assembly.
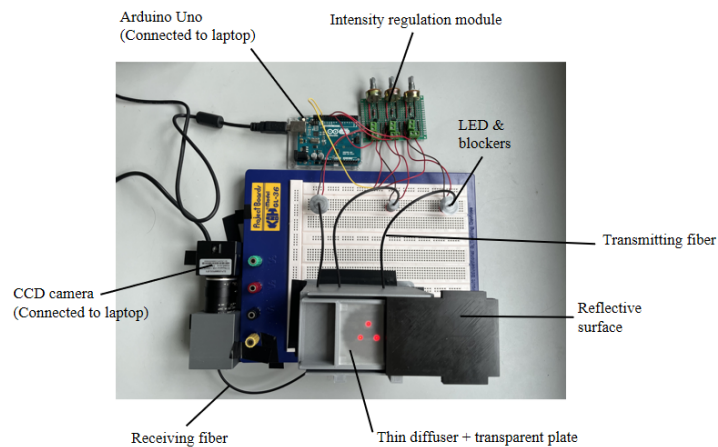

Figure A.21: Experimental platform of prototype

Based on the proposed illumination pattern and mechanical structure, the experimental platform on a sin-

gle set of fiber transceivers has been constructed, where the micro-controller unit Arduino together with the intensity regulation module enables independent control over different light sources. The thinner diffuser is adopted to reduce the amount of light reflected by the diffuser itself, thereby increasing the intensity difference.

The relationship between light intensities and the edge positions in the horizontal or vertical direction with 0.25 mm steps is illustrated in Figure A.22. Each curve corresponds to the optical fiber marked by the circle in the same color while the black curve represents the case where all transmitting fibers are illuminated. In the case of negative edge distance, the receiving fiber at the center is covered by the reflective surface. Compared to the previous measurement result in Figure A.13, more stabilized light intensity levels can be observed with a smooth transition and over doubled intensity difference. In addition, the light intensities do not strictly conform to the superposition principle where there is a difference between the sum of measured intensities from individual light sources and the intensity when all light sources are turned on.
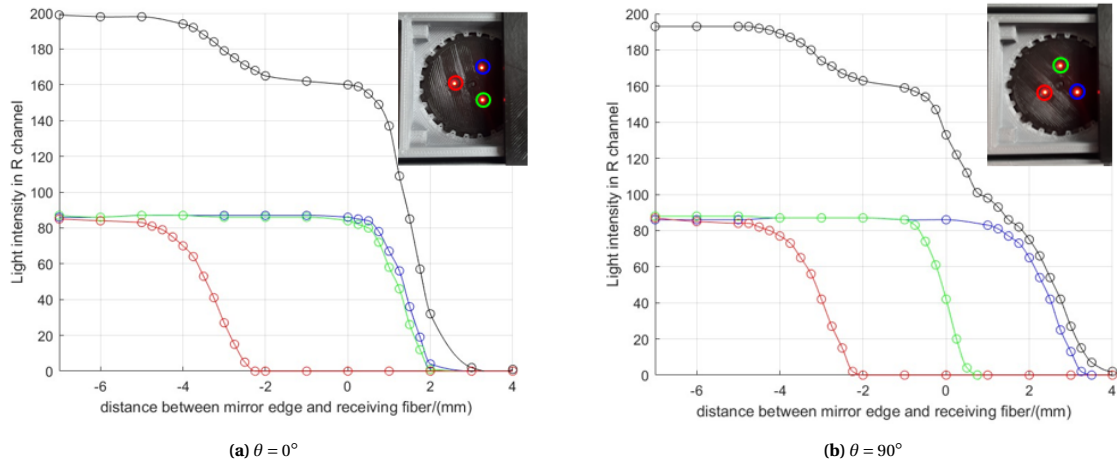

(a) $\theta = 0°$

(b) $\theta = 90°$

Figure A.22: Intensity curves measured by experimental platform

A.4.3. Planar position decoupling algorithm

Figure A.23a depicts the preliminary estimation method for the orientation of the object, represented as the blue rectangle. Among the receiving fibers in the vicinity of the object's boundaries, the four farthest fiber coordinates from the average center are identified, along with annotations in the figure. Coordinates identified as corners are connected sequentially to infer the approximate positions of the object's edges, with exact expressions denoted by purple lines. These lines are used to classify the remaining receiving fibers along different edges, illustrated as dots in distinct colors. The orientation of each edge, indicated by the slope of the green lines, is obtained using a linear regression model over the corresponding set of fiber coordinates. Geometric constraints, such as the orthogonality of neighboring edges, are considered to determine the orientation of the object.
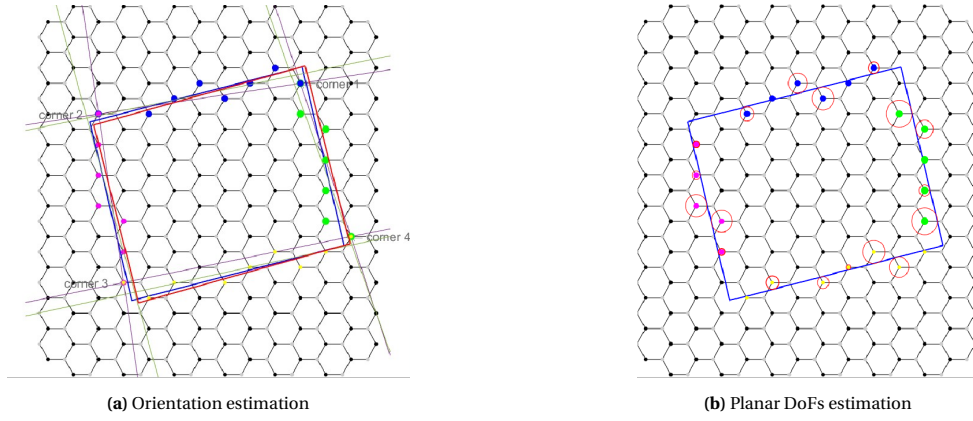
(a) Orientation estimation **(b)** Planar DoFs estimation

Figure A.23: Planar position decoupling by simulated fiber optic sensor array

When an object is arbitrarily positioned on the simulated sensing array, with a density of roughly 2000 receiving fibers per square meter and a ratio of 10 between the size of the edge and the radius of the inscribed circle in a hexagonal array, the inaccuracy for initial orientation estimation can be limited to below 0.05 radians. Once the appropriate mathematical expression between light intensities and edge distance at a certain orientation is identified, an optimization problem can be formulated to refine the planar degrees of freedom by determining the optimal object position tangential to a set of circles centered around receiving fibers with radii equal to the edge distance.

A.5. chronological summary of literature studies

Table A.5: A chronological summary of planar position measurement systems

Year	Ref.	Sensor Type	Motion Stroke	Resolution	Accuracy	Application
1994	[6]	Capacitive sensor	$X/Y:(85/60)$mm	$XY:0.1\mu$m	$XY:\pm6.5\mu$m	Wafer stepper
2003	[15]	Optical encoder	$XY:35$mm	$XY:0.2\mu$m	$XY:\pm2.5\mu$m	Precision stage systems
			θ_z : arc-seconds	$\theta_z:\simeq5\mu$rad		
2003	[37]	Photodiodes	$XY:1$mm	$XY:0.0635$mm	$XY:\simeq\pm75\mu$m	Drift monitoring
2007	[49]	Interferometer	$XY:20$mm	$X/Y:(0.6/0.3)$nm	$XY:\simeq\pm1$nm	Ultra-precision systems
			$\theta_z:\pm1\mu$rad	$\theta_z:\simeq0.04\mu$rad		
2008	[57]	Interferometer	$XY:120$mm	$XY:20$nm	$XY:\simeq\pm10$nm	Wafer stepper
			θ_z : micro-radians	θ_z : submicro-radians		
2010	[29]	Photodiodes	$X/Y:(8.67/13.03)$mm	$X/Y:(27.4/38.7)$nm	$XY:4.1\mu$m	Precision stage systems
2010	[20]	Inductive sensor	$XY:30$mm	$XY:10\mu$m	$XY:\simeq\pm10\mu$m	Precise machine tools
2010	[24]	Capacitive sensor	$XY:300\mu$m	$XY:0.4$nm	$XY:\simeq\pm0.25\mu$m	Probe devices
2011	[46]	Optical encoder	$XY:20$mm	$XY:10$nm	$XY:\pm0.5\mu$m	Micro-machine tools
			θ_z : arc-seconds			
2012	[56]	Interferometer	$XY:\simeq100$mm	$XY:80$nm	$XY:\pm20\mu$m	Precision stage systems
			θ_z : micro-radians			
2012	[25]	Photodiodes	$XY:9$mm	$XY:\simeq6\mu$m	$XY:50\mu$m	Micro-robotics
2013	[59]	Capacitive sensor	$XY:20$mm	$XY:0.308\mu$m	$XY:\simeq\pm0.25$mm	Microscopes
2013	[5]	Optical encoder	$XY:\simeq10$mm	$XY:1$nm	$XY:\pm2.9\mu$m	Nano-positioning stages
2015	[35]	Magnetic sensor	$XY:\simeq40$mm	$XY:5\mu$m	$XY:\simeq\pm0.2$mm	Planar motor
2015	[62]	Magnetic sensor	$XY:\pm2$mm	$XY:1\mu$m	$XY:\pm6\mu$m	Precision stage systems
			$\theta_z:\pm0.3°$	$\theta_z:6\mu$rad	$\theta_z:\simeq\pm40\mu$rad	
2015	[33]	Optical encoder	$XY:12$mm	Not given	$X/Y:\pm(0.5/1.5)\mu$m	Precision stage systems
			$\theta_z:\simeq\pm1$mrad		$\theta_z:\simeq\pm3\mu$rad	
2016	[58]	Capacitive sensor	$XY:16$mm	$XY:0.1\mu$m	$XY:\simeq\pm10\mu$m	Microscopes
2016	[1]	Photodiodes	$XY:2$mm	$XY:3.1\mu$m	$XY:\pm9.7\mu$m	Microscopy stages
2016	[8]	Magnetic sensor	$XY:50$mm	$XY:\simeq0.1\mu$m	$XY:\simeq\pm0.5$mm	Precision stage systems
			$\theta_z:\pm15°$	$\theta_z:\simeq10\mu$rad		
2016	[50]	Interferometer	$XY:50$mm	$XY:10$nm	$X/Y:\pm(41/36)$nm	Nano-positioning stages
2016	[22]	Photodiodes	$X:200$mm	$XY:0.1\mu$m	$X:\pm37\mu$m	Wafer stages
				$\theta_z:1\mu$rad		
2016	[32]	Photodiodes	$XY:9$mm	Not given	$XY:\pm0.2\mu$m	Planar stages
			θ_z : unlimited		$\theta_z:\pm0.15\mu$rad	
2017	[51]	Optical encoder	$X/Y:(140/50)$mm	$XY:\pm2.44$nm	$XY:\pm1\mu$m	Planar precision stages
			$\theta_z:\pm0.25°$			
2017	[60]	Interferometer	$XY:300$mm	$XY:0.1\mu$m	$X:\pm0.36\mu$m	Precision stage systems
			$\theta_z:\pm10°$	$\theta_z:\simeq0.6\mu$rad	$\theta_z:1.7\mu$rad	
2018	[31]	Optical encoder	$XY:16$mm	$XY:25$nm	$XY:\pm0.51\mu$m	Precision stage systems
			$\theta_z:\pm45°$	$\theta_z:0.001°$		
2019	[53]	Inductive sensor	$XY:20$mm	$XY:2.67\mu$m	$XY:\simeq\pm0.2$mm	Precision stage systems
2019	[12]	Photodiodes	$XY:200$mm	$XY:0.094$mm	$XY:0.623$mm	Transport systems
2020	[3]	Optical encoder	$XY:\simeq20$mm	$XY:\simeq1$nm	$X/Y:\pm(0.96/0.77)$nm	Precision stage systems
			θ_z : unlimited	$\theta_z:4\mu$rad	$\theta_z:\simeq\pm4\mu$rad	
2020	[23]	Interferometer	$XY:20$mm	$XY:0.15$nm	$X/Y:\pm(60/80)$nm	Planar motion stages
			$\theta_z:\pm50\mu$rad	$\theta_z:\simeq4\mu$rad	$\theta_z:\pm600\mu$arcs	
2020	[54]	Inductive sensor	$XY:20.8$mm	$XY:0.3\mu$m	$XY:\simeq\pm0.4$mm	Planar stages
2020	[30]	Optical encoder	$XY:24$mm	$XY:0.1\mu$m	$XY:\pm3.1\mu$m	Precision stage systems
			θ_z : unlimited	$\theta_z:0.005°$	$\theta_z:\pm0.02°$	
2021	[26]	Photodiodes	$XY\theta_z$	Not given	Not given	Microscopes
2021	[4]	Photodiodes	$XY:\simeq7.2$mm	$XY:\simeq35\mu$m	$XY:\simeq\pm0.6$mm	Air bearing systems
			θ_z : unlimited		$\theta_z:\simeq\pm4.8°$	
2022	[9]	Optical encoder	$XY:200\mu$m	$XY:\simeq2$nm	$XY:\pm2$nm	Microscopes
			$\theta_z:\pm0.5$mrad	$\theta_z:\simeq1\mu$rad	$\theta_z:\simeq\pm1\mu$rad	
2022	[21]	Photodiodes	$XY:100$mm	$XY:0.1\mu$m	$X/Y:\pm(12.4/12.9)\mu$m	Microscopes
					$\theta_z:\pm3.1$mrad	
2022	[34]	Inductive sensor	$XY:20$mm	Not given	$XY:\simeq\pm0.3$mm	Three-axis machines
2022	[36]	Photodiodes	$XY:\simeq22$mm	$XY:\simeq35\mu$m	Not given	Transport systems
			θ_z : unlimited			
2023	[63]	Optical encoder	$XY:15$mm	$XY:0.2\mu$m	$XY:\pm0.1\mu$m	Precision stage systems
2023	[40]	Capacitive sensor	$XY:200$mm	$XY:\simeq0.5\mu$m	$X/Y:\pm(8.2/6.8)\mu$m	Precision stage systems
2023	[16]	Magnetic sensor	$XY:\simeq26$mm	$XY:0.15\mu$m	$XY:\simeq\pm50\mu$m	Micro-robotics

A.6. Calendar schedule

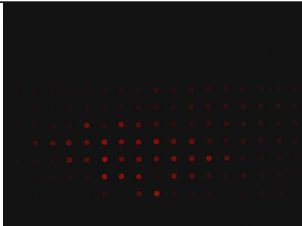		Nov				Dec				Jan				Feb				Mar				Apr				May				Jun		
	6	13	20	27	4	11	18	25	1	8	15	22	29	5	12	19	26	4	11	18	25	1	8	15	22	29	6	13	20	27	3	10	17	24
Mathematical modelling of sensing medium																																		
Build up simulation platform of distributed fiber optic sensors																																		
Mechanical design and setup of prototype with single set of fibers																																		
Build up the electric circuit in light generation module																																		
Christmas break																																		
Prepare for literature review and project proposal																																		
Design the mechanical structure and subsystems of smart surface																																		
Assembly of smart surface																																		
Apply and verify the 2D position measurement algorithm in dark environment																																		
Apply image recognition algorithm in detecting planar object and fiber states																																		
Build up receiving fiber classifier under light and obstruction interference																																		
Evaluate the performance of distributed fiber optic sensors																																		
Finish graduation thesis and prepare for final defense																																		

☐ Simulation   ☐ Experiment   ☐ Vacation   ☐ Thesis preparation

Figure A.24: Calendar schedule for research
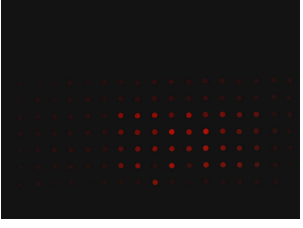
B
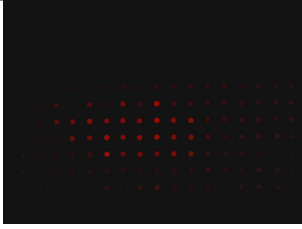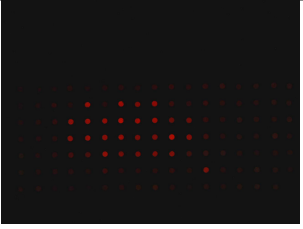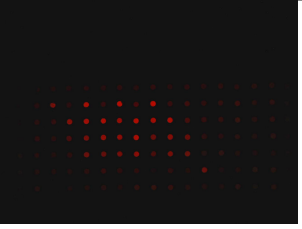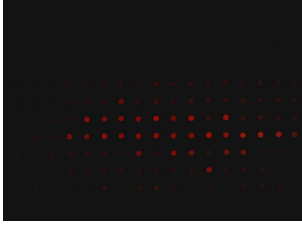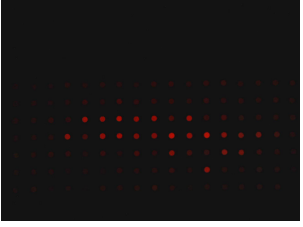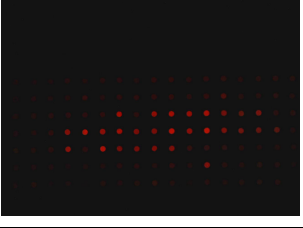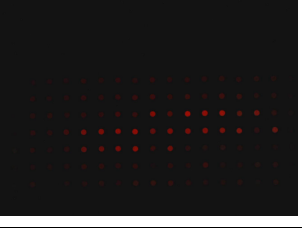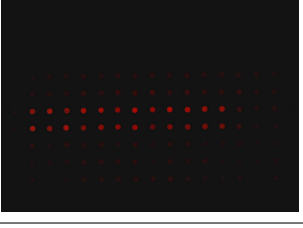
Measurement samples for sensor surface

This section lists the raw images and the corresponding outputs by planar positioning algorithms for all test samples used for system evaluation. During the actual installation of the fiber array, two pairs of receiving fibers are placed in reverse positions. The first pair of fibers appear in the image in column 9, rows 6 and 7, and the second pair of fibers appear in the image in column 12, rows 5 and 6. This can be compensated by swapping the actual fiber positions stored in database D_X in the program.
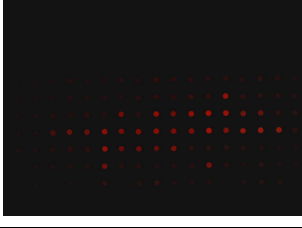
Table B.1: Raw images of the receiving fiber array for each measurement sample

No.	Illumination cases		
	$G_T = 1$	$G_T = 2$	$G_T = 3$
1			
2			
3			

No.	Illumination cases		
	$G_T = 1$	$G_T = 2$	$G_T = 3$
4			
5			
6			
7			
8			
9			
10			

No.	Illumination cases		
	$G_T = 1$	$G_T = 2$	$G_T = 3$
11			
12			
13			
14			
15			
16			
17			
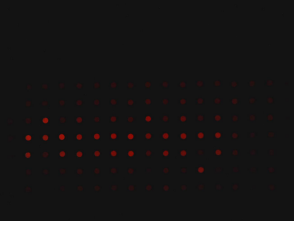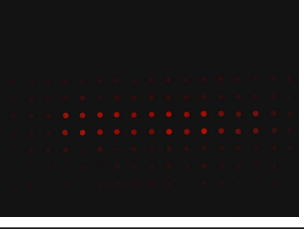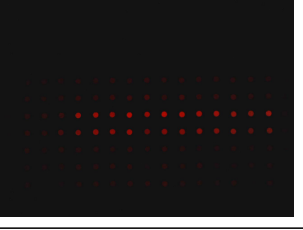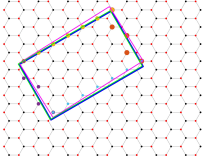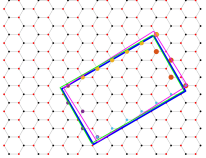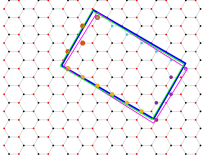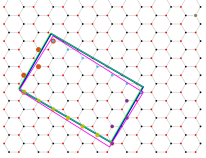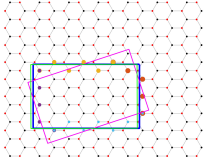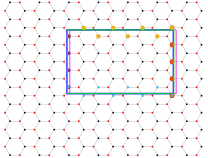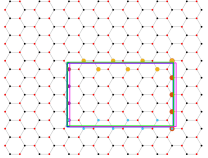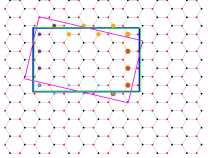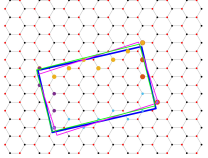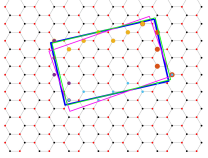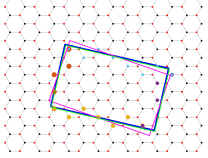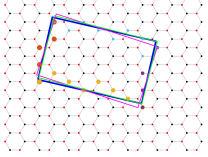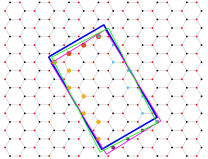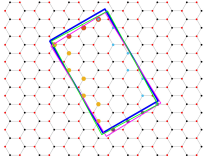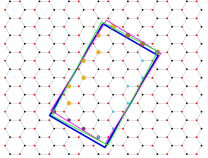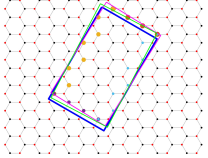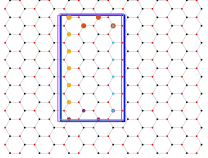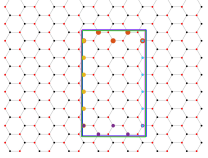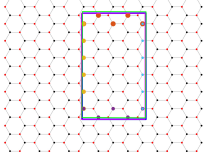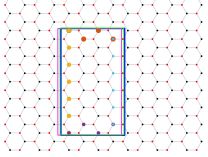
No.	Illumination cases		
	$G_T = 1$	$G_T = 2$	$G_T = 3$
18			
19			
20			

Table B.2: Measurement results for each measurement sample

No.	Real position (blue rectangle)			Initial estimation (magenta rectangle)			Optimized solution (green rectangle)			Figure
	x_0	y_0	α	$\overline{x_B}$	$\overline{y_B}$	α_0	x_c	y_c	α_c	
1	-20	12	0.524	-19.86	15.20	0.558	-20.82	12.35	0.524	
2	20	-12	0.524	21.71	-8.80	0.522	18.19	-11.47	0.515	
3	20	12	2.605	21.71	8.80	2.620	18.61	11.36	2.616	
4	-20	-12	2.605	-19.86	-15.20	2.538	-20.47	-11.65	2.618	

No.	Real position (blue rectangle)			Initial estimation (magenta rectangle)			Optimized solution (green rectangle)			Figure
	x_0	y_0	α	$\overline{x_B}$	$\overline{y_B}$	α_0	x_c	y_c	α_c	
5	-16	-16	0	-14.01	-16.00	0.201	-18.65	-15.94	0.002	
6	16	16	0	18.48	16	0	16.77	15.99	0.002	
7	16	-16	0	18.48	-16	0	15.95	-15.33	0.005	
8	-16	16	0	-18.48	16	-0.229	-15.23	16.03	-0.005	
9	-6	-12	0.227	-5.25	-11.27	0.312	-7.04	-10.49	0.245	
10	6	12	0.227	5.94	10.29	0.325	7.52	12.53	0.242	
11	6	-12	2.915	-5.88	-12.57	2.807	6.34	-13.21	2.927	
12	-6	12	2.915	-5.88	11.64	2.829	-5.49	13.29	2.882	

No.	Real position (blue rectangle)			Initial estimation (magenta rectangle)			Optimized solution (green rectangle)			Figure
	x_0	y_0	α	$\overline{x_B}$	$\overline{y_B}$	α_0	x_c	y_c	α_c	
13	-0.5	-7.4	2.094	2.31	-12.00	2.121	0.93	-10.16	2.043	
14	0.5	7.4	2.094	2.31	-4.00	1.021	1.26	5.58	2.064	
15	0.5	-7.4	1.047	2.31	-4	1.021	1.27	-5.34	1.083	
16	-0.5	7.4	1.047	2.31	12	1.021	0.55	10.62	1.087	
17	-10	8	1.571	-12.78	8.00	1.571	-10.92	9.48	1.573	
18	10	-8	1.571	9.24	-8.00	1.571	9.82	-8.52	1.568	
19	10	8	1.571	9.24	8.00	1.571	9.96	9.48	1.566	
20	-10	-8	1.571	-12.77	-8.00	1.571	-10.67	-7.67	1.573	

C

MATLAB codes

C.1. Generation of fiber optic sensor array in hexagonal cells

This function establishes a simulated fiber optic sensor array and returns the position of receiving fibers near the object boundaries, as well as the position lists of all transmitting fibers and receiving fibers by the inputs of sensor surface parameters and object information. The function only plots the sensor array when input argument `plotbase` is true. Other custom functions are also listed.

```matlab
function [boundaries, dict_nbf, Ts, Rs] = sensor_array_v2A5(rc, rr, h, w, angle, cp, ...
    plotbase)
dy=2*rc;
dx=rc*sqrt(3);
A = 1:7;
A=pi/3*A;
ts = [];
rs = [];
boundaries = [];
dict_nbf = struct();
for yk = [0:dy:2*w,0:-dy:-2*w]
    for xk = [0:dx:2*h,0:-dx:-2*h]
        xp = xk;
        yp = sqrt(3)*xk/3 + yk;
        if -105 < xp - 2*rc/sqrt(3) && xp + 2*rc/sqrt(3) < 105  && -74 <yp - rc && ...
            yp  + rc < 74
            T = [xp+1i*yp]+rc*exp(1i*A)*2/sqrt(3);
            Vertx = real(T);
            Verty = imag(T);
            Vts = [Vertx(1:2:6);Verty(1:2:6)];
            Vrs = [Vertx(2:2:6);Verty(2:2:6)];
            ts = horzcat(ts, Vts);
            rs = horzcat(rs, Vrs);
            if plot_base
                plot(T,Color= [0.7 0.7 0.7]);
                hold on;
            end
        end
    end
end
ts(1,:) = round(ts(1,:),4);
ts(2,:) = round(ts(2,:),4);
ts = unique(ts','rows');
nts = size(ts,1);
rs(1,:) = round(rs(1,:),4);
rs(2,:) = round(rs(2,:),4);
rs = unique(rs','rows');
nrs = size(rs,1);
elements_column = unique(ts(:,2));
elements_column = flipud(elements_column);
Ts1 = [];
Ts2 = [];
Ts3 = [];
```

```matlab
42  Rs1 = [];
43  Rs2 = [];
44  Rs3 = [];
45  for i = 1:size(elements_column,1)
46      index = find(ts(:,2) == elements_column(i));
47      index1 = find(rs(:,2) == elements_column(i));
48      t_index  = ts(index,:);
49      r_index  = rs(index1,:);
50      if mod(i,3) == 1
51          Ts1 = vertcat(Ts1, t_index);
52          Rs1 = vertcat(Rs1, r_index);
53      elseif mod(i,3) == 2
54          Ts2 = vertcat(Ts2, t_index);
55          Rs2 = vertcat(Rs2, r_index);
56      else
57          Ts3 = vertcat(Ts3, t_index);
58          Rs3 = vertcat(Rs3, r_index);
59      end
60  end
61  Ts = {Ts1, Ts2, Ts3};
62  Rs = {Rs1, Rs2, Rs3};
63  [xw, yw, fun_b] = panel_state(h, w, angle, cp);
64  if plotbase == true
65      set(gcf, 'Position', [0 0 400 300]);
66      axis equal;
67      xlim([-105 105]);
68      ylim([-74 74]);
69      scatter(ts(:,1),ts(:,2),5,[1,0,0],'filled');
70  %     scatter(Ts1(:,1),Ts1(:,2), 5,[0.7,0.7,0.7],'filled');
71  %     scatter(Ts3(:,1),Ts3(:,2), 12,[1,0,0],'filled');
72  %     scatter(Ts2(:,1),Ts2(:,2), 5,[0.7,0.7,0.7],'filled');
73      scatter(rs(:,1),rs(:,2),5,[0,0,0],'filled');
74  %     plot([xw,xw(1)], [yw,yw(1)],'b',LineWidth = 2);
75      axis off;
76      set(gca,'looseInset',[0 0 0 0]);
77  else
78      Rdmax = 91;
79      Rdmin = 2;
80      sigma = 0.3295;
81      Imax = max_I(rc, rr, rr, sigma, false)/3;
82      funb = @(x,y) fun_b{1,1}(x,y) <= 0 & fun_b{1,2}(x,y) <= 0 & fun_b{1,3}(x,y) <= 0 & ...
             fun_b{1,4}(x,y) <= 0;
83      lightspot = [];
84      real_boundaries = [];
85      for seq = 1:3
86          for seqr = 1:3
87              rs = Rs{1, seqr};
88              RF_query = struct();
89              for i = 1:size(rs,1)
90                  str_rf = encode(rs(i,1), rs(i,2));
91                  x1 = rs(i,1) + 2*rc/sqrt(3); y1 = rs(i,2);
92                  x2 = rs(i,1) - rc/sqrt(3); y2 = rs(i,2) + rc;
93                  x3 = rs(i,1) - rc/sqrt(3); y3 = rs(i,2) - rc;
94                  if seq == seqr && ismember([round(x1,2), ...
                        round(y1,2)],round(Ts{1,seq},2),'rows')
95                      fun = @(x,y) exp(-((x - x1).^2 + (y - ...
                            y1).^2)/(2*(rr*sigma)^2)).*((x-rs(i,1)).^2 + (y-rs(i,2)).^2 <= ...
                            rr^2).*funb(x,y).*exp(-((x - rs(i,1)).^2 + (y - ...
                            rs(i,2)).^2)/(2*(rr*sigma)^2));
96                      RF_query.RF_case = 1;
97                  elseif seqr == mod(seq, 3) + 1 && ismember([round(x2,2), ...
                        round(y2,2)],round(Ts{1,seq},2),'rows')
98                      fun = @(x,y) exp(-((x - x2).^2 + (y - ...
                            y2).^2)/(2*(rr*sigma)^2)).*((x-rs(i,1)).^2 + (y-rs(i,2)).^2 <= ...
                            rr^2).*funb(x,y).*exp(-((x - rs(i,1)).^2 + (y - ...
                            rs(i,2)).^2)/(2*(rr*sigma)^2));
99                      RF_query.RF_case = 2;
100                 elseif seq == mod(seqr, 3) + 1 && ismember([round(x3,2), ...
                        round(y3,2)],round(Ts{1,seq},2), 'rows')
101                     fun = @(x,y) exp(-((x - x3).^2 + (y - ...
                            y3).^2)/(2*(rr*sigma)^2)).*((x-rs(i,1)).^2 + (y-rs(i,2)).^2 <= ...
```

```
                                    rr^2).*funb(x,y).*exp(-((x - rs(i,1)).^2 + (y - ...
                                    rs(i,2)).^2)/(2*(rr*sigma)^2));
102                         RF_query.RF_case = 3;
103                     else
104                         fun = @(x,y) x.*0 + y.*0;
105                     end
106                     I0 = integral2(fun, rs(i,1) -rr, rs(i,1) + rr, rs(i,2) -rr, rs(i,2) + ...
                            rr);
107                     if I0/Imax ≥ 0.05
108                         RF_query.I = R_int(I0/Imax, Rdmax, Rdmin);
109                         if I0/Imax > 0.95
110                             RF_query.I = Rdmax;
111                         else
112                             real_boundaries = [real_boundaries;[rs(i,1), rs(i,2)]];
113                         end
114                         if ismember(str_rf, fieldnames(dict_bf)) == 0
115                             Seq_query = struct();
116                             lightspot = [lightspot;[rs(i,1), rs(i,2)]];
117                         else
118                             Seq_query = dict_bf.(str_rf);
119                         end
120                         Seq_query.(strcat('s',num2str(seq))) = RF_query;
121                         dict_bf.(str_rf) = Seq_query;
122                     end
123                 end
124             end
125     end
126     boundaries_index = boundary(lightspot(:,1), lightspot(:,2));
127     boundaries = [lightspot(boundaries_index(1:end-1),1), ...
                    lightspot(boundaries_index(1:end-1),2)];
128     real_boundaries = unique(real_boundaries,"rows");
129     if size(real_boundaries, 1) ≠ 0
130         boundaries = union(boundaries, real_boundaries,'rows');
131     end
132     dict_nbf = struct();
133     for i = 1:size(boundaries, 1)
134         str_xy = encode(boundaries(i,1), boundaries(i,2));
135         seq_keys = {'s1','s2','s3'};
136         Seq_query = dict_bf.(str_xy);
137         for j = 1:size(seq_keys,2)
138             if ismember(seq_keys{1, j}, fieldnames(Seq_query)) == 0
139                 for k = 1: size(Rs, 2)
140                     if ismember(boundaries(i,:), Rs{1,k},'rows')
141                         if j == k
142                             RF_case = 1;
143                         elseif k == mod(j, 3) + 1
144                             RF_case = 2;
145                         elseif j == mod(k, 3) + 1
146                             RF_case = 3;
147                         end
148                         Seq_query.(seq_keys{1,j}) = struct('RF_case', RF_case, 'I', 0);
149                         break
150                     end
151                 end
152             end
153         end
154         dict_nbf.(str_xy) = Seq_query;
155     end
156 end
```

```
1  function [xw, yw, f] = panel_state(h,w,angle,cp)
2      x1 = cp(1) + 1/2 * (h*cos(angle) - w*sin(angle));
3      y1 = cp(2) + 1/2 * (h*sin(angle) + w*cos(angle));
4      x2 = cp(1) + 1/2 * (-h*cos(angle) - w*sin(angle));
5      y2 = cp(2) + 1/2 * (-h*sin(angle) + w*cos(angle));
6      x3 = cp(1) + 1/2 * (-h*cos(angle) + w*sin(angle));
7      y3 = cp(2) + 1/2 * (-h*sin(angle) - w*cos(angle));
8      x4 = cp(1) + 1/2 * (h*cos(angle) + w*sin(angle));
9      y4 = cp(2) + 1/2 * (h*sin(angle) - w*cos(angle));
```

```matlab
10      xw = [x1, x2, x3, x4];
11      yw = [y1, y2, y3, y4];
12      f = cell(1,4);
```

```matlab
1  function str_xy = encode(x,y)
2      str_xy = strcat('x',num2str(x),'y',num2str(y));
3      str_xy = strrep(str_xy,'.','_');
4      str_xy = strrep(str_xy,'-','n');
```

```matlab
1  function Imax = max_I(rc, rt, rr, sigma)
2      x0 = 0; y0 = 0;
3      x1 = x0 + 2*rc/sqrt(3); y1 = y0;
4      x2 = x0 - rc/sqrt(3); y2 = y0 + rc;
5      x3 = x0 - rc/sqrt(3); y3 = y0 - rc;
6      fun1 = @(x,y) exp(-((x - x1).^2 + (y - y1).^2)/(2*(rt*sigma)^2)).*exp(-((x - ...
            x0).^2 + (y - y0).^2)/(2*(rr*sigma)^2)).*((x-x0).^2 + (y-y0).^2 <= rr^2);
7      fun2 = @(x,y) exp(-((x - x2).^2 + (y - y2).^2)/(2*(rt*sigma)^2)).*exp(-((x - ...
            x0).^2 + (y - y0).^2)/(2*(rr*sigma)^2)).*((x-x0).^2 + (y-y0).^2 <= rr^2);
8      fun3 = @(x,y) exp(-((x - x3).^2 + (y - y3).^2)/(2*(rt*sigma)^2)).*exp(-((x - ...
            x0).^2 + (y - y0).^2)/(2*(rr*sigma)^2)).*((x-x0).^2 + (y-y0).^2 <= rr^2);
9      val1 = integral2(fun1, x0 -rr, x0 + rr, y0 -rr, y0 + rr);
10     val2 = integral2(fun2, x0 -rr, x0 + rr, y0 -rr, y0 + rr);
11     val3 = integral2(fun3, x0 -rr, x0 + rr, y0 -rr, y0 + rr);
12     Imax = val1 + val2 + val3;
```

```matlab
1  function [R] = R_int(Ir, dmax, dmin)
2      cmax = 1;
3      cmin = 0;
4      R = round((Ir - cmin) * (dmax - dmin) / (cmax - cmin) + dmin);
```

C.2. Snapshot, light switching program, and brightness calculation

The code in this subsection corresponds to brightness evaluations of a single set of transmitting and receiving fibers in Chapter 3.

```matlab
1  a = arduino('COM3','Uno');
2  %% light switching program, the last argument 1 represents off and 0 represents on
3  writeDigitalPin(a, 'D11', 1);
4  writeDigitalPin(a, 'D10', 1);
5  writeDigitalPin(a,  'D9', 1);
6  %%
7  R_value = [];
8  video1 = webcam(1);
9  video1.BacklightCompensation = 1;
10 video1.Brightness = 0;
11 video1.Contrast = 32;
12 video1.WhiteBalanceMode = "manual";
13 video1.WhiteBalance = 4900;
14 video1.ExposureMode= "manual";
15 video1.Exposure = -2;
16 video1.Sharpness = 0;
17 timer_id = timer;
18 timer_id.StartDelay = 1;
19 timer_id.Period = 0.1;
20 timer_id.ExecutionMode = 'fixedSpacing';
21 timer_id.TasksToExecute = 5;
22 maxiter = timer_id.TasksToExecute;
23 timer_id.TimerFcn = {@timer_handler, video1, maxiter};
24 start(timer_id);
25 state = true;
26 while state == true
27     pause(0.1);
28     if timer_id.TasksExecuted == timer_id.TaskstoExecute
29         state = false;
30     end
31 end
```

```matlab
32  delete(timer_id);
33  load('R.mat');
34  mean_R = round(mean(R));
35  if max(R) ≠ 0
36      mean_R = round(mean(R(R≠0)));
37  end
```

```matlab
1   function timer_handler(obj,event,cam,maxiter)
2       persistent R_value
3       img = snapshot(cam);
4       imshow(img);
5       [centers, radii] = imfindcircles(img, [18 30]);
6       if size(centers,1) == 0
7           centers = [845, 721];
8           radii = 20;
9       end
10      R_value = [R_value, RDetection(img, centers(1), centers(2), radii)];
11      if size(R_value,2) == maxiter
12          R = [R_value];
13          save('R.mat', 'R');
14      end
```

```matlab
1   function R = RDetection(img, x, y, r)
2       [X,Y] = meshgrid(1:size(img,2), 1:size(img,1));
3       filter = (X − x).^2 + (Y − y).^2 ≤ 0.5*r.^2;
4       R_channel = img(:,:,1);
5       R_filtered = R_channel(filter);
6       R = round(mean(R_filtered));
```

C.3. Corner recognition algorithm

```matlab
1   function [corner_pts, hori_state] =  find_corners_v3(boundaries, h, w)
2       ecp = [mean(boundaries(:,1)),mean(boundaries(:,2))];
3       max_x = boundaries(find(boundaries(:,1) == max(boundaries(:,1))), :);
4       max_y = boundaries(find(boundaries(:,2) == max(boundaries(:,2))), :);
5       min_x = boundaries(find(boundaries(:,1) == min(boundaries(:,1))), :);
6       min_y = boundaries(find(boundaries(:,2) == min(boundaries(:,2))), :);
7       if min([size(max_x, 1), size(max_y,1), size(min_x,1), size(min_y,1)]) ≥ 3 || ...
            max([size(max_x, 1), size(max_y,1), size(min_x,1), size(min_y,1)]) ≥ 5
8           hori_state = true;
9           xuyu = union(max_x(find(max_x(:,2) == max(max_x(:,2)))), ...
                    :),max_y(find(max_y(:,1) == max(max_y(:,1))), :) ,'rows');
10          xlyu = union(min_x(find(min_x(:,2) == max(min_x(:,2)))), ...
                    :),max_y(find(max_y(:,1) == min(max_y(:,1))), :) ,'rows');
11          xlyl = union(min_x(find(min_x(:,2) == min(min_x(:,2)))), ...
                    :),min_y(find(min_y(:,1) == min(min_y(:,1))), :) ,'rows');
12          xuyl = union(max_x(find(max_x(:,2) == min(max_x(:,2)))), ...
                    :),min_y(find(min_y(:,1) == max(min_y(:,1))), :) ,'rows');
13
14          xuyu = boundaries(find(boundaries(:,1) ≥ min(xuyu(:,1)) & ...
                                boundaries(:,1) ≤ max(xuyu(:,1)) & ...
15                               boundaries(:,2) ≥ min(xuyu(:,2)) & ...
16                               boundaries(:,2) ≤ max(xuyu(:,2))), :)
17
18
19          xlyu = boundaries(find(boundaries(:,1) ≥ min(xlyu(:,1)) & ...
                                boundaries(:,1) ≤ max(xlyu(:,1)) & ...
20                               boundaries(:,2) ≥ min(xlyu(:,2)) & ...
21                               boundaries(:,2) ≤ max(xlyu(:,2))), :)
22
23
24          xlyl = boundaries(find(boundaries(:,1) ≥ min(xlyl(:,1)) & ...
                                boundaries(:,1) ≤ max(xlyl(:,1)) & ...
25                               boundaries(:,2) ≥ min(xlyl(:,2)) & ...
26                               boundaries(:,2) ≤ max(xlyl(:,2))), :)
27
28          xuyl = boundaries(find(boundaries(:,1) ≥ min(xuyl(:,1)) & ...
                                boundaries(:,1) ≤ max(xuyl(:,1)) & ...
29                               boundaries(:,2) ≥ min(xuyl(:,2)) & ...
30                               boundaries(:,2) ≥ min(xuyl(:,2)) & ...
```

```matlab
31                                          boundaries(:,2) <= max(xuyl(:,2)))), :)
32          corner_pts = [xuyu(1,:); xlyu(1,:); xlyl(1,:); xuyl(1,:)];
33          max_area = 0;
34          for i = 1:size(xuyu,1)
35              for j = 1:size(xlyu,1)
36                  for k = 1: size(xlyl, 1)
37                      for l = 1:size(xuyl, 1)
38                          selected_set = [xuyu(i,:); xlyu(j,:); xlyl(k,:); xuyl(l,:)];
39                          area = polyarea(selected_set(:,1), selected_set(:,2));
40                          if area > max_area
41                              corner_pts = selected_set;
42                              max_area = area;
43                          end
44                      end
45                  end
46              end
47          end
48          if max(boundaries(:,1)) - min(boundaries(:,1)) < max(boundaries(:,2)) - ...
                 min(boundaries(:,2))
49              corner_pts = [corner_pts(2,:); ...
                     corner_pts(3,:);corner_pts(4,:);corner_pts(1,:)];
50          end
51      else
52          hori_state = false;
53          corner_pts = [max_x(1,:); max_y(1,:); min_x(1,:); min_y(1,:)];
54          max_area = 0;
55          for i = 1:size(max_x,1)
56              for j = 1:size(max_y,1)
57                  for k = 1:size(min_x,1)
58                      for l = 1:size(min_y,1)
59                          selected_set = [max_x(i,:); max_y(j,:); min_x(k,:); min_y(l,:)];
60                          area = polyarea(selected_set(:,1), selected_set(:,2));
61                          if area > max_area
62                              corner_pts = selected_set;
63                              max_area = area;
64                          end
65                      end
66                  end
67              end
68          end
69      end
70      eh =  (norm(corner_pts(1,:) - corner_pts(2,:)) + norm(corner_pts(3,:) - ...
             corner_pts(4,:)))/2;
71      ew =  (norm(corner_pts(1,:) - corner_pts(4,:)) + norm(corner_pts(2,:) - ...
             corner_pts(3,:)))/2;
72      if eh < ew
73          corner_pts = [corner_pts(end,:); corner_pts(1:end-1,:)];
74          new_column = (1:size(corner_pts,1))';
75          corner_pts = [corner_pts, new_column];
76          if corner_pts(1,1) + corner_pts(4,1) == corner_pts(2,1) + corner_pts(3,1)
77              theta_axis = pi/2;
78          else
79              theta_axis = atan((corner_pts(1,2) + corner_pts(4,2) - corner_pts(2,2) - ...
                     corner_pts(3,2))/(corner_pts(1,1) + corner_pts(4,1) - corner_pts(2,1) ...
                     - corner_pts(3,1)));
80          end
81          transformed_corner_pts = corner_pts*[cos(theta_axis), -sin(theta_axis), 0; ...
                 sin(theta_axis), cos(theta_axis), 0; 0, 0, 1];
82          tp2s = [];
83          for i = 1: size(transformed_corner_pts,1)
84              o_x = transformed_corner_pts(i,1) - mean(transformed_corner_pts(:,1));
85              o_y = transformed_corner_pts(i,2) - mean(transformed_corner_pts(:,2));
86              radpt = atan2(o_y,o_x);
87              if theta_axis < 0
88                  radpt = mod(radpt + pi, 2*pi);
89              end
90              if radpt < 0
91                  radpt = radpt + 2*pi;
92              end
93              tp2s = [tp2s;radpt];
94          end
```

```matlab
95            transformed_corner_pts = sortrows([transformed_corner_pts,tp2s],4);
96            corner_pts_order = transformed_corner_pts(:,3);
97            corner_pts = corner_pts(corner_pts_order',1:2);
98       end
```

C.4. Edge classification algorithm

```matlab
1       function [C_dict] = edge_classifier_v4(boundaries, ac_pts, rc)
2       f = cell(1,size(ac_pts,1));
3       nf = size(f,2);
4       for i = 1:nf
5           nc = e2cg(i, nf); % find all corner numbers cotained in a given edge number
6           f{1,i} = @(x,y) (ac_pts(nc(2),2) - ac_pts(nc(1),2))/norm(ac_pts(nc(2),:) - ...
                ac_pts(nc(1),:))*(x - ac_pts(nc(1),1)) ...
7               - (ac_pts(nc(2),1) - ac_pts(nc(1),1))/norm(ac_pts(nc(2),:) - ...
                  ac_pts(nc(1),:))*(y- ac_pts(nc(1),2));
8       end
9       C = cell(1,nf);
10      corner_pts = [];
11      for i = 1: size(boundaries,1)
12          L_norm = [];
13          for j = 1: nf
14              dis = abs(f{1,j}(boundaries(i,1), boundaries(i,2)));
15              L_norm = vertcat(L_norm, [dis,j]);
16          end
17          L_norm = sortrows(L_norm, 1);
18          if L_norm(2,1) - L_norm(1,1) ≤ rc
19              min_ind = [L_norm(1,2), L_norm(2,2)];
20              nc = eg2c(L_norm(1,2), L_norm(2,2)); % find the corner number by given ...
                    edge numbers
21              corner_pts = vertcat(corner_pts, [boundaries(i,:), nc]);
22          else
23              min_ind = [L_norm(1,2)];
24          end
25          for k = min_ind
26              C{1,k} = [C{1,k}; boundaries(i,:)];
27          end
28      end
29      rest_corners = corner_pts(¬ismember(corner_pts(:,1:2), ac_pts,'rows'), :);
30      if size(rest_corners,1) > 0
31          for m = 1: size(rest_corners,1)
32              index = rest_corners(m,3); % corner number of rest corner
33              cornerm = ac_pts(index,:);
34              adj_index = sort([setdiff(e2cg(index, nf), index), setdiff(c2eg(index, ...
                    nf), index)]);
35              exc_class = false;
36              cancel_group = [];
37              adj_dis = double.empty(0,2);
38              r_pro = double.empty(0,2);
39              for n = 1: size(adj_index,2)
40                  index_adj = adj_index(1,n);
41                  cornern = ac_pts(index_adj,:);
42                  adj_dis(n) = abs((cornerm(1,2) - cornern(1,2))/norm(cornerm - ...
                        cornern)*(rest_corners(m,1) - cornern(1,1)) ...
43                                    - (cornerm(1,1) - cornern(1,1))/norm(cornerm - ...
                                      cornern)*(rest_corners(m,2) - cornern(1,2)));
44                  r_pro_dis = sqrt(norm(rest_corners(m,1:2) - cornern)^2 - adj_dis(n)^2);
45                  pro_dis = norm(cornerm - cornern);
46                  r_pro(n) = pro_dis - r_pro_dis;
47                  if r_pro_dis > pro_dis
48                      exc_class = true;
49                      cancel_group = [cancel_group, cg2e(index, index_adj)];
50                  end
51              end
52              if exc_class == false
53                  if abs(r_pro(1) - r_pro(2)) < 1e-4
54                      selected_group = NaN;
55                  elseif r_pro(1) > r_pro(2)
56                      selected_group = cg2e(index, adj_index(1,1));
```

```
57                  else
58                      selected_group = cg2e(index, adj_index(1,2));
59                  end
60                  cancel_group = setdiff(c2eg(index, nf), selected_group);
61              end
62              for cg = 1:size(cancel_group, 2)
63                  row_indices = find(ismember(C{1, ...
                        cancel_group(1,cg)},rest_corners(m,1:2),'rows'));
64                  logical_index = true(size(C{1, cancel_group(1,cg)},1),1);
65                  logical_index(row_indices) = false;
66                  C{1, cancel_group(1,cg)} = C{1, cancel_group(1,cg)}(logical_index,:);
67              end
68          end
69      end
70      C_dict = struct();
71      for m = 1: nf
72          str_xy = strcat('Edge', num2str(m));
73          C_dict.(str_xy) = C{1,m};
74      end
```

```
1   function ne = eg2c(nc1, nc2)
2       x1 = min(nc1, nc2);
3       x2 = max(nc1, nc2);
4       if x2 - x1 == 1
5           ne = x1;
6       else
7           ne = x2;
8       end
```

```
1   function [nc] = e2cg(ne, n)
2       if ne == 1
3           nc(1) = ne;
4           nc(2) = n;
5       else
6           nc(1) = ne - 1;
7           nc(2) = ne;
8       end
```

```
1   function [nc] = e2cg(ne, n)
2       if ne == 1
3           nc(1) = ne;
4           nc(2) = n;
5       else
6           nc(1) = ne - 1;
7           nc(2) = ne;
8       end
```

```
1   function ne = cg2e(nc1, nc2)
2       x1 = min(nc1, nc2);
3       x2 = max(nc1, nc2);
4       if x2 - x1 == 1
5           ne = x2;
6       else
7           ne = x1;
8       end
```

C.5. Initial orientation estimation

The code in this subsection corresponds to Equations from 4.11 to 4.14 in subsection 4.4.

```
1   function [C_new, new_corner_pts, alpha] = edge_function_v3(C, h, w, corner_pts, ...
        hori_state)
2       f = cell(1,4);
```

```matlab
3       alphas = [];
4       C_str = fieldnames(C);
5       for i = 1: size(C_str,1)
6           c_str = C_str{i,1};
7           edgei = C.(c_str);
8           meanxi = mean(edgei(:,1));
9           meanyi = mean(edgei(:,2));
10          Lxx = round(sum((edgei(:,1) - meanxi).^2),4);
11          Lxy = round(sum((edgei(:,1) - meanxi).*(edgei(:,2) - meanyi)),4);
12          if Lxx == 0
13              alpha = pi/2;
14          else
15              alpha = asin(Lxy/sqrt(Lxx^2 + Lxy^2));
16          end
17          alphas = [alphas, alpha];
18          f{1,i} = @(x,y) sin(alpha)*(x-meanxi) - cos(alpha)*(y-meanyi);
19      end
20      fun = @(x) (min(abs(x - alphas(2)), pi - abs(x - alphas(2))))^2 + (min(abs(x - ...
            alphas(4)), pi - abs(x - alphas(4))))^2 +...
21      ((min(abs(x - pi/2 - alphas(1)), pi - abs(x - pi/2 - alphas(1))))^2 + (min(abs(x ...
            - pi/2 - alphas(3)), pi - abs(x - pi/2 - alphas(3))))^2) .*(x>0) +...
22      ((min(abs(x + pi/2 - alphas(1)), pi - abs(x + pi/2 - alphas(1))))^2 + (min(abs(x ...
            + pi/2 - alphas(3)), pi - abs(x + pi/2 - alphas(3))))^2) .*(x≤0);
23      [alpha, mindiff] = fminbnd(fun, -pi/2, pi/2);
24      C_new = cell(1,4);
25      if hori_state == true
26          if abs(alpha + pi/2) ≤ pi/12
27              alpha = alpha + pi;
28          end
29      end
30      if hori_state == false
31          if alpha < 0
32              if h > w
33                  alpha = alpha + pi;
34              elseif h == w
35                  alpha = alpha + pi/2;
36              end
37          end
38          Rm = [cos(alpha), -sin(alpha), 0; sin(alpha), cos(alpha), 0; 0, 0, 1];
39          new_column = (1:size(corner_pts,1))';
40          corner_pts = [corner_pts, new_column];
41          corner_pts_R = corner_pts * Rm;
42          tp2s = [];
43          for i = 1: size(corner_pts_R, 1)
44              o_x = corner_pts_R(i,1) - mean(corner_pts_R(:,1));
45              o_y = corner_pts_R(i,2) - mean(corner_pts_R(:,2));
46              radpt = atan2(o_y, o_x);
47              if radpt < 0
48                  radpt = radpt + 2*pi;
49              end
50              tp2s = [tp2s;radpt];
51          end
52          transformed_corner_pts = sortrows([corner_pts_R,tp2s],4);
53          corner_pts_order = transformed_corner_pts(:,3);
54          new_corner_pts = corner_pts(corner_pts_order',1:2);
55          for i = 1: size(C_new, 2)
56              nc = e2cg(i, 4);
57              nc(1) = transformed_corner_pts(nc(1),3);
58              nc(2) = transformed_corner_pts(nc(2),3);
59              ne = cg2e(nc(1), nc(2));
60              str_nc = strcat('Edge', num2str(ne));
61              C_new{1, i} = C.(str_nc);
62          end
63      else
64          for i = 1: 4
65              Edgestr = strcat('Edge',num2str(i));
66              C_new{1,i} = C.(Edgestr);
67          end
68          new_corner_pts = corner_pts;
69      end
```

C.6. Global position estimation for simulated platform

```matlab
1   rc = 8;
2   rr = 7.2;
3   h = 100;
4   w = 60;
5   angle = 30*pi/180; % orientation of panel
6   cp = [-20,12];
7   [boundaries,dict_nfcf, Ts, Rs] = sensor_array_v2A5(rc,rr,h,w,angle,cp,false);
8   [corner_pts,hori_state] = find_corners_v3(boundaries, h, w);
9   [C_list] = edge_classifier_v4(boundaries,corner_pts, rc);
10  [C, corner_pts, alpha] = edge_function_v3(C_list, h, w, corner_pts,hori_state);
11  fg1 = @(xi, yi, x) cos(x(3))*(xi - x(1)) + sin(x(3))*(yi - x(2)) - h/2;
12  fg2 = @(xi, yi, x) -sin(x(3))*(xi - x(1)) + cos(x(3))*(yi - x(2)) - w/2;
13  fg3 = @(xi, yi, x) -cos(x(3))*(xi - x(1)) - sin(x(3))*(yi - x(2)) - h/2;
14  fg4 = @(xi, yi, x) sin(x(3))*(xi - x(1)) - cos(x(3))*(yi - x(2)) - w/2;
15  fg = {fg1, fg2, fg3, fg4};
16  lb = [-h, -w, -pi];
17  ub = [h, w, pi];
18  Aeq = []; beq = []; A = []; b = [];
19  Rdmax = 91;
20  Rdmin = 2;
21  iter_opt = true;
22  iter_num = 0;
23  x0 = [mean(boundaries(:,1)), mean(boundaries(:,2)), alpha];
24  while iter_opt && iter_num ≤ 5
25      Edge = cell(1,4);
26      Edge_fc = cell(1,4);
27      for i = 1:size(boundaries,1)
28          str_rf = encode(boundaries(i,1), boundaries(i,2));
29          Seq_query = dict_nfcf.(str_rf);
30          Seq_keys = fieldnames(Seq_query);
31          Edge_type = 0;
32          if ismember(boundaries(i,:), corner_pts,'rows') == 0
33              % not corner points
34              for j = 1: size(C,2)
35                  if ismember(boundaries(i,:), C{1,j}, 'rows') == 1
36                      Edge_type = j;
37                      break;
38                  end
39              end
40              if Edge_type == 0
41                  continue;
42              end
43          else
44              continue;
45          end
46          for j = 1: size(Seq_keys,1)
47              RF_query = Seq_query.(Seq_keys{j});
48              seq = str2num(Seq_keys{j}(2));
49              alpha0 = alpha + (Edge_type - 1)*pi/2;
50              if RF_query.RF_case == 1
51                  orientation = alpha0;
52              elseif RF_query.RF_case == 2
53                  if alpha0 ≤ 2*pi/3
54                      orientation = 2*pi/3 - alpha0;
55                  else
56                      orientation = 8*pi/3 - alpha0;
57                  end
58              elseif RF_query.RF_case == 3
59                  if alpha0 ≤ 4*pi/3
60                      orientation = 4*pi/3 - alpha0;
61                  else
62                      orientation = 10*pi/3 - alpha0;
63                  end
64              end
65              if RF_query.I == Rdmax
66                  distance = edge_distance(0.9, orientation, rc, rr);
67                  Edge_fc{1, Edge_type} = [Edge_fc{1, Edge_type}; [boundaries(i,1), ...
                      boundaries(i,2), distance,1]];
```

```
68              elseif RF_query.I == 0
69                  distance = edge_distance(0.1, orientation, rc, rr);
70                  Edge_fc{1, Edge_type} = [Edge_fc{1, Edge_type}; [boundaries(i,1), ...
                        boundaries(i,2), distance,-1]];
71              else
72                  distance = edge_distance(R_int(RF_query.I, Rdmax, Rdmin), ...
                        orientation, rc, rr);
73                  Edge{1, Edge_type} = [Edge{1, Edge_type}; [boundaries(i,1), ...
                        boundaries(i,2), distance]];
74              end
75          end
76      end
77      [design_parameters, val] = fmincon(@(x)fun(x,Edge,fg), x0, A, b, Aeq, beq, lb, ...
            ub, @(x)nonlcon(x,Edge_fc,fg),[]);
78      x0 = design_parameters;
79      if abs(design_parameters(1,3) - alpha) < 0.01
80          iter_opt = false;
81      end
82      alpha = design_parameters(1,3);
83      iter_num = iter_num + 1;
84  end
```

```
1  function d = edge_distance(I_r, ori, rc, rr)
2      p1 = 2.367*(rc/rr)^2 - 3.894*(rc/rr) + 1.744;
3      p2 = 2.233*(rc/rr)^2 - 2.989*(rc/rr) + 8.391;
4      p3 = 0.2032*(rc/rr)^2 - 0.8997*(rc/rr) + 0.1378;
5      a = p1*cos(2*ori) + p2;
6      b = p3*cos(ori);
7      d = rr*(1/a*log(1/I_r - 1) + b);
```

```
1  function f = fun(x, Edge, fg)
2      f = 0;
3      for i = 1:size(Edge,2)
4          Edgei = Edge{1,i};
5          for j = 1: size(Edgei,1)
6              f = f + (fg{1,i}(Edgei(j,1), Edgei(j,2), x) - Edgei(j,3))^2;
7          end
8      end
```

```
1  function [c, ceq] = nonlcon(x, Edge_fc, fg)
2      ceq = [];
3      c = [];
4      for i = 1:size(Edge_fc,2)
5          Edge_fci = Edge_fc{1,i};
6          for j = 1: size(Edge_fci,1)
7              c = [c,Edge_fci(j,4)*(fg{1,i}(Edge_fci(j,1),Edge_fci(j,2),x) - Edge_fci(j,3))];
8          end
9      end
```

C.7. Global position estimation for image inputs

```
1  rc = 8;
2  rr = 7.2;
3  h = 100;
4  w = 60;
5  nx = 17;
6  ny = 7;
7  imagepath = 'D:\Msc thesis\Graduation Project\MATLAB code\fiberarray\samplepos1\';
8  nocover = imread([imagepath 'nocover.jpg']); % Figure 6.2a
9  T1max = imread([imagepath 'T1max.jpg']); % Figure 5.13a
10 T1min = imread([imagepath 'T1min.jpg']); % Figure 5.13d
11 T2max = imread([imagepath 'T2max.jpg']); % Figure 5.13b
12 T2min = imread([imagepath 'T2min.jpg']); % Figure 5.13e
13 T3max = imread([imagepath 'T3max.jpg']); % Figure 5.13c
14 T3min = imread([imagepath 'T3min.jpg']); % Figure 5.13f
```

```matlab
15   Sample1 = {};
16   for j = 1:3
17       Sample1{1,j} = imread([imagepath strcat('s1','T',num2str(j),'.jpg')]); % Figure ...
             6.3d-f
18   end
19   %%Database Dx
20   [centers, radii] = imfindcircles(nocover, [12 30]);
21   [cls_x,cx] = kmeans(centers(:,1),nx);
22   [cls_y,cy]= kmeans(centers(:,2),ny);
23   M_cls = [cls_x, cls_y];
24   M_s2u_x = sortrows([cx,(1:nx)'], 1);
25   c_s2u_x = M_s2u_x(:,2);
26   M_s2u_y = sortrows([cy,(1:ny)'], 1);
27   c_s2u_y = M_s2u_y(:,2);
28   U = []; V = [];
29   uR = []; vR = []; rR = [];
30   xR = []; yR = [];
31   for j = 1: size(c_s2u_y,1)
32       for i = 1: size(c_s2u_x, 1)
33           [xw, yw] = im2worldpos_tf(i, j, nx, ny, rc);
34           idim = find(M_cls(:,1) == c_s2u_x(i) & M_cls(:,2) == c_s2u_y(j));
35           xdim = centers(idim,1);
36           ydim = centers(idim,2);
37           radii_im = radii(idim);
38           U = [U; i];
39           V = [V; j];
40           uR = [uR; xdim];
41           vR = [vR; ydim];
42           rR = [rR; radii_im];
43           xR = [xR; xw];
44           yR = [yR; yw];
45       end
46   end
47   Dx = table(U, V, uR, vR, rR, xR, yR);
48   xc1 = Dx{(Dx.U == 12 & Dx.V == 5), 'xR'};
49   yc1 = Dx{(Dx.U == 12 & Dx.V == 5), 'yR'};
50   xc2 = Dx{(Dx.U == 12 & Dx.V == 6), 'xR'};
51   yc2 = Dx{(Dx.U == 12 & Dx.V == 6), 'yR'};
52   xc3 = Dx{(Dx.U == 9 & Dx.V == 6), 'xR'};
53   yc3 = Dx{(Dx.U == 9 & Dx.V == 6), 'yR'};
54   xc4 = Dx{(Dx.U == 9 & Dx.V == 7), 'xR'};
55   yc4 = Dx{(Dx.U == 9 & Dx.V == 7), 'yR'};
56   Dx{(Dx.U == 12 & Dx.V == 5), 'xR'} = xc2;
57   Dx{(Dx.U == 12 & Dx.V == 6), 'xR'} = xc1;
58   Dx{(Dx.U == 12 & Dx.V == 5), 'yR'} = yc2;
59   Dx{(Dx.U == 12 & Dx.V == 6), 'yR'} = yc1;
60   Dx{(Dx.U == 9 & Dx.V == 6), 'xR'} = xc4;
61   Dx{(Dx.U == 9 & Dx.V == 7), 'xR'} = xc3;
62   Dx{(Dx.U == 9 & Dx.V == 6), 'yR'} = yc4;
63   Dx{(Dx.U == 9 & Dx.V == 7), 'yR'} = yc3;
64   %% Database Dp
65   U = []; V = [];
66   GR = []; GT = [];
67   PL = []; PU = [];
68   T1 = {T1max, T1min};
69   T2 = {T2max, T2min};
70   T3 = {T3max, T3min};
71   T = {T1, T2, T3};
72   for j = 1: size(c_s2u_y,1)
73       for i = 1: size(c_s2u_x, 1)
74           rows = Dx{(Dx.U == i & Dx.V == j),['uR','vR','rR']};
75           for k = 1: size(T,2)
76               Tk = T{1,k};
77               Bu = RDetection(Tk{1,1}, rows(1), rows(2), rows(3));
78               Bl = RDetection(Tk{1,2}, rows(1), rows(2), rows(3));
79               GT = [GT;k];
80               GR = [GR; mod(i-1,3) + 1];
81               PL = [PL; Bl];
82               PU = [PU; Bu];
83               U = [U; i];
84               V = [V; j];
```

```matlab
85              end
86          end
87  end
88  Dp = table(U, V, GT, GR, PL, PU);
89  %% Read samples
90  check_sample = Sample1;
91  h = 100; w = 60; rc = 8; rr = 7.2;
92  angle = 30*pi/180; cp = [-20,12];
93  grouped_spot = cell(1,3);
94  real_boundaries = [];
95  U = []; V = []; GT = [];
96  t = []; P = []; N = [];
97  for seq = 1: size(check_sample,2)
98      imagei = check_sample{1,seq};
99      for i = 1 : nx
100         for j = 1: ny
101             rowx = Dx{(Dx.U == i & Dx.V == j),['uR','vR','rR']};
102             rowp = Dp{(Dp.U == i & Dp.V == j & Dp.GT == seq),['PL','PU','GR']};
103             I_obs = RDetection(imagei, rowx(1), rowx(2), rowx(3));
104             if rowp(3) == seq
105                 RF_case = 1;
106             elseif rowp(3) == mod(seq,3) + 1
107                 RF_case = 2;
108             elseif seq == mod(rowp(3),3) + 1
109                 RF_case = 3;
110             end
111             if I_obs ≤ rowp(1)
112                 n_obs = 0;
113             elseif I_obs ≥ rowp(2)
114                 n_obs = 1;
115             else
116                 n_obs = (I_obs - rowp(1))/(rowp(2) - rowp(1));
117             end
118             U = [U;i];
119             V = [V;j];
120             GT = [GT; seq];
121             t = [t; RF_case];
122             P = [P; I_obs];
123             N = [N; n_obs];
124         end
125     end
126 end
127 %% Fiber selection by ball pivoting algorithm
128 DM = table(U,V,GT,t,P,N);
129 UF = Dp{(Dp.PU - Dp.PL ≥ 30),['U','V','GT']};
130 SI = unique(DM{(ismember([DM.U, DM.V, DM.GT], UF,'rows') & DM.N ≥ ...
131     0.1),['U','V','GT']},'rows');
131 lightspot = unique(Dx{ismember([Dx.U, Dx.V],SI(:,1:2),'rows'),['xR', 'yR']},'rows');
132 bc = BallConcave(lightspot); % see reference [19]
133 radius = 12;
134 boundaries = bc.GetConcave_Ball(radius);
135 boundaries = unique(boundaries,'rows');
136 SB = unique(Dx{ismember([Dx.xR, Dx.yR], boundaries,'rows'),['U','V']},"rows");
137 %% Planar positioning algorithms
138 [corner_pts, hori_state] = find_corners_v3(boundaries, h, w, rr);
139 C_list = edge_classifier_v4(boundaries,corner_pts, rr);
140 [C, new_corner_pts, alpha] = edge_function_v3(C_list, h, w, corner_pts, hori_state);
141 Pts_size = [40, 30, 20, 10];
142 C1c = [0.8500 0.3250 0.0980];
143 C2c = [0.9290 0.6940 0.1250];
144 C3c = [0.4940 0.1840 0.5560];
145 C4c = [0.3010 0.7450 0.9330];
146 Cxc = [C1c; C2c; C3c; C4c];
147 Cir_color = [1 0 0; 0 1 0; 0 1 1];
148 figure; hold on;
149 axis off
150 sensor_array_v2A5(rc, rr, h, w, angle, cp, true);
151 for i = 1: size(C,2)
152     scatter(C{1,i}(:,1), C{1,i}(:,2),Pts_size(i),Cxc(i,:),'filled');
153 end
154 [xest1, yest1] = panel_state(h,w,alpha,[mean(boundaries(:,1)),mean(boundaries(:,2))]);
```

```matlab
155  plot([xest1,xest1(1)], [yest1,yest1(1)],'m',LineWidth = 1);
156  fg1 = @(xi, yi, x) cos(x(3))*(xi - x(1)) + sin(x(3))*(yi - x(2)) - h/2;
157  fg2 = @(xi, yi, x) -sin(x(3))*(xi - x(1)) + cos(x(3))*(yi - x(2)) - w/2;
158  fg3 = @(xi, yi, x) -cos(x(3))*(xi - x(1)) - sin(x(3))*(yi - x(2)) - h/2;
159  fg4 = @(xi, yi, x) sin(x(3))*(xi - x(1)) - cos(x(3))*(yi - x(2)) - w/2;
160  fg = {fg1, fg2, fg3, fg4};
161  lb = [-105, -74, -pi];
162  ub = [105, 74, pi];
163  Aeq = []; beq = []; A = []; b = [];
164  iter_opt = true;
165  iter_num = 0;
166  x0 = [mean(boundaries(:,1)), mean(boundaries(:,2)), alpha]
167  while iter_opt && iter_num ≤ 5
168      Edge = cell(1,4);
169      Edge_fc = cell(1,4);
170      Edge_val = cell(1,4);
171      for i = 1: size(C,2)
172          Ci = C{1,i};
173          for j = 1:size(Ci,1)
174              if ismember(Ci(j,:), new_corner_pts, 'rows') == 0
175                  rowx = Dx{(Dx.xR == Ci(j,1) & Dx.yR  == Ci(j,2)),['U','V']};
176                  for k = 1:3
177                      alphap = alpha + (i-1)*pi/2;
178                      rown = DM{(DM.U == rowx(1) & DM.V == rowx(2) & DM.GT == ...
179                          k),['t','N']};
179                      if rown(1) == 1
180                          theta = alphap;
181                      elseif rown(1) == 2
182                          if alphap ≤ 2*pi/3
183                              theta = 2*pi/3 - alphap;
184                          else
185                              theta = 8*pi/3 - alphap;
186                          end
187                      elseif rown(1) == 3
188                          if alphap ≤ 4*pi/3
189                              theta = 4*pi/3 - alphap;
190                          else
191                              theta = 10*pi/3 - alphap;
192                          end
193                      end
194                      if rown(2) ≥ 0.9
195                          distance = edge_distance(0.9, theta, rc, rr);
196                          Edge_fc{1, i} = [Edge_fc{1, i}; [Ci(j,1), Ci(j,2), distance,1]];
197                      elseif rown(2) ≤ 0.1
198                          distance = edge_distance(0.1, theta, rc, rr);
199                          Edge_fc{1, i} = [Edge_fc{1, i}; [Ci(j,1), Ci(j,2), distance,-1]];
200                      else
201                          distance = edge_distance(rown(2), theta, rc, rr);
202                          Edge{1, i} = [Edge{1, i}; [Ci(j,1), Ci(j,2), distance]];
203                      end
204                  end
205              end
206          end
207      end
208      [design_parameters, val] = fmincon(@(x)fun(x,Edge,fg), x0, A, b, Aeq, beq, lb, ...
209          ub, @(x)nonlcon(x,Edge_fc,fg),[]);
209      x0 = design_parameters;
210      if abs(design_parameters(1,3) - alpha) < 0.01
211          iter_opt = false;
212      end
213      alpha = design_parameters(1,3);
214      iter_num = iter_num + 1;
215  end
216  [xest, yest] = panel_state(h,w,design_parameters(3),[design_parameters(1), ...
        design_parameters(2)]);
217  plot([xest,xest(1)], [yest,yest(1)],'g',LineWidth = 1);
```

```matlab
1  function [x, y] = im2worldpos_tf(idx, idy, nx, ny, rc)
2      y = -rc*(idx - (nx + 1)/2);
```

```matlab
3      if mod(idx,2) == 1
4          x = rc/sqrt(3)*(6*(idy - (ny + 1)/2) + 2);
5      else
6          x = rc/sqrt(3)*(6*(idy - (ny + 1)/2) - 1);
7      end
8      x = round(x,4);
9      y = round(y,4);
```

References

[1]   Alejandro Alvarez-Aguirre et al. "Performance improvement of optical mouse sensors: Application in a precision planar stage". In: *2016 International Conference on Manipulation, Automation and Robotics at Small Scales (MARSS)*. IEEE. 2016, pp. 1–6.

[2]   Pizzagalli Amandine and Giusti Gaël. *Thinning equipment technology and market trends for semiconductor device*. YDR20067. Yole Développement (Yole), June 2020.

[3]   Antoine N André et al. "Sensing one nanometer over ten centimeters: A microencoded target for visual in-plane position measurement". In: *IEEE/ASME Transactions on Mechatronics* 25.3 (2020), pp. 1193–1201.

[4]   Fritz Arnout. "Distributed optical sensor surface concept for planar object detection: Using optical fibers as a vision distributor". In: (2021).

[5]   Shorya Awtar and Gaurav Parmar. "Design of a large range XY nanopositioning system". In: *Journal of Mechanisms and Robotics* 5.2 (2013), p. 021008.

[6]   MHW Bonse, F Zhu, and JW Spronck. "A new two-dimensional capacitive position transducer". In: *Sensors and Actuators A: Physical* 41.1-3 (1994), pp. 29–32.

[7]   Kahina Boutoustous et al. "Distributed control architecture for smart surfaces". In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. 2010, pp. 2018–2024.

[8]   Longmin Chen et al. "A position measurement method for 3-degree-of-freedom motion based on magnetic detection". In: *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering* 230.5 (2016), pp. 467–479.

[9]   Qinrui Cheng, Ting Xu, and Peisen Huang. "Systematic error analysis and parameter design of a vision-based phase estimation method for ultra-precision positioning". In: *Optics Express* 30.5 (2022), pp. 8330–8344.

[10]   *CMOS image sensor*. https://thinklucid.com/tech-briefs/understanding-digital-image-sensors/.

[11]   Wikipedia contributors. *Atan2 — Wikipedia, The Free Encyclopedia*. https://en.wikipedia.org/wiki/Atan2. Accessed: 2024-07-11. 2024.

[12]   J.F. Elfferich et al. "The simulation and design of a one-sided distributed optical sensor for wafer tracking". In: (2019).

[13]   *Fiber optic sensors*. http://www.mtiinstruments.com/technology-principles/fiber-optic-sensors.

[14]   W Gao et al. "Measurement technologies for precision positioning". In: *CIRP annals* 64.2 (2015), pp. 773–796.

[15]   Wei Gao et al. "A surface motor-driven planar motion stage integrated with an XYθZ surface encoder for precision positioning". In: *Precision Engineering* 28.3 (2004), pp. 329–337.

[16]   Gabriel Géron et al. "Characterization of a magnetic localization method based on Hall effect sensor array for microrobot position tracking". In: *Journal of Micro-Bio Robotics* (2023), pp. 1–13.

[17]   Ichinose Go. *Exposure apparatus and device manufacturing method*. KR20170133526A. Korean Intellectual Property Office, Nov. 2019.

[18]   Kenneth TV Grattan and T Sun. "Fiber optic sensor technology: an overview". In: *Sensors and Actuators A: Physical* 82.1-3 (2000), pp. 40–61.

[19]   Jumanco hide. *The third way of thinking - ball pivoting*. https://www.cnblogs.com/chnhideyoshi/p/ConcaveHull.html. Mar. 2014.

[20]   Yousef Hojjat et al. "Development of an inductive encoder for simultaneous measurement of two-dimensional displacement". In: *The International Journal of Advanced Manufacturing Technology* 53 (2011), pp. 681–688.

[21] Rico HT Hooijschuur et al. "Air-based contactless wafer precision positioning system: Contactless sensing using charge coupled devices". In: *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology* 237.6 (2023), pp. 1276–1288.

[22] Tao Huang et al. "Integrated robust tracking controller design for a developed precision planar motor with equivalent disturbances". In: *IET Control Theory & Applications* 10.9 (2016), pp. 1009–1017.

[23] Wei-Lun Huang et al. "Integrating time-optimal motion profiles with position control for a high-speed permanent magnet linear synchronous motor planar motion stage". In: *Precision Engineering* 68 (2021), pp. 106–123.

[24] Xinghui Huang et al. "Nano-positioning of an electromagnetic scanner with a MEMS capacitive sensor". In: *Mechatronics* 20.1 (2010), pp. 27–34.

[25] Ioan Alexandru Ivan, Mihai Ardeleanu, and Guillaume J Laurent. "High dynamics and precision optical measurement using a position sensitive detector (PSD) in reflection-mode: Application to 2D object tracking over a smart surface". In: *Sensors* 12.12 (2012), pp. 16771–16784.

[26] Binghui Jia. "The spatial modulation characteristic of optical fiber array for linear and angle displacement precision measurement". In: *Global Intelligent Industry Conference 2020*. Vol. 11780. SPIE. 2021, pp. 433–438.

[27] D.Tobiason Joseph. *Encoder using fiber optic receiver channels*. EP1475613B1. European Patent Office, Oct. 2004.

[28] D.Tobiason Joseph. *Optical path array and angular filter for translation and orientation sensing*. US70 75097B2. United States Patent and Trademark Office, May 2007.

[29] A Khiat et al. "Two-dimension fiber optic sensor for high-resolution and long-range linear measurements". In: *Sensors and Actuators A: Physical* 158.1 (2010), pp. 43–50.

[30] Jong-Ahn Kim et al. "Absolute planar position and full angle orientation sensor using a 2-D color-coded binary scale". In: *IEEE Transactions on Instrumentation and Measurement* 69.5 (2019), pp. 2225–2231.

[31] Jong-Ahn Kim et al. "Note: An absolute XY-Θ position sensor using a two-dimensional phase-encoded binary scale". In: *Review of Scientific Instruments* 89.4 (2018).

[32] Stefan Lampaert et al. "XY360–Planar Positioning Stage with a PSD sensor and ferrofluid bearings". In: *DSPE Conference on Precision Mechatronics*. 2016, p. 57.

[33] Yongmeng Liu et al. "Use of two planar gratings to measure 3-DOF displacements of planar moving stage". In: *IEEE Transactions on Instrumentation and Measurement* 64.1 (2014), pp. 163–169.

[34] Kunlong Ma et al. "A New 2D Displacement Measurement Method Based on an Eddy Current Sensor and Absolute Encoding". In: *Machines* 10.11 (2022), p. 1077.

[35] Keir Maguire. "Multi-degree of freedom position sensor for planar motors". PhD thesis. University of British Columbia, 2015.

[36] van der Wielen Matthijs. "Development of an edge detection sensor for planar objects: Using multiple light colours and optical fibres as a distributed vision sensor". In: (2022).

[37] Tuck Wah Ng. "The optical mouse as a two-dimensional displacement sensor". In: *Sensors and Actuators A: Physical* 107.1 (2003), pp. 21–25.

[38] *Optical fiber*. https://www.keyence.eu/ss/products/sensor/sensorbasics/fiber/info/.

[39] Edmund Optics. *1000μm with 1 Fiber, Optical Grade Plastic Light Guide*. https://www.edmundoptics.com/f/optical-grade-fiber-optics/11460/.

[40] Kai Peng et al. "Planar Two-Dimensional Capacitive Displacement Sensor Based on Time Grating". In: *IEEE Transactions on Industrial Electronics* (2023).

[41] Dr. Simon Philipps. *Photovoltaics Report*. Fraunhofer ISE and Werner Warmuth, PSE Projects GmbH, May 2024.

[42] Kelly Pickerel. *Big-wafer solar panels arent quite ready for their residential debut*. https://www.solarpowerworldonline.com/2020/08/big-wafer-solar-panels-arent-quite-ready-for-their-residential-debut/. Aug. 2020.

[43] *Power devices*. https://www.alpha-powers.com/technology-sic-characteristics.

[44] Multicomp Pro. *5mm Round LED Lamp*. https://www.farnell.com/datasheets/2863877.pdf. Sept. 2020.

[45] Pyrasied. *Extruded acrylic light diffusers*. https://shop.pyrasied.nl/en/product-category/light-diffusers/extruded-acrylic-light-diffusers/. Accessed: 2024-04-08.

[46] Seung-Kook Ro and Jong-Kweon Park. "A compact ultra-precision air bearing stage with 3-DOF planar motions using electromagnetic motors". In: *International Journal of Precision Engineering and Manufacturing* 12 (2011), pp. 115–119.

[47] R. Munnig Schmidt et al. *The Design of High Performance Mechatronics: High-Tech Functionality by Multidisciplinary System Integration*. 3rd. NLD: IOS Press, 2020, pp. 504–505. ISBN: 978-1-64368-051-4.

[48] ON Semiconductor. *NPN General - Purposed Amplifier 2N3904*. https://www.onsemi.com/pdf/datasheet/2n3904-d.pdf. July 2021.

[49] Hidenori Shinno, Hayato Yoshioka, and K Taniguchi. "A newly developed linear motor-driven aerostatic XY planar motion table system for nano-machining". In: *CIRP annals* 56.1 (2007), pp. 369–372.

[50] Marta Torralba et al. "Design optimization for the measurement accuracy improvement of a large range nanopositioning stage". In: *Sensors* 16.1 (2016), p. 84.

[51] Kai Treichel. "Modeling and robust adaptive tracking control of a planar precision positioning system". PhD thesis. Dissertation, Ilmenau, TU Ilmenau, 2017, 2017.

[52] HP Vuong. "Air-based contactless actuation system for thin substrates: The concept of using a controlled deformable surface". PhD thesis. Delft University of Technology, 2016.

[53] Liang Wu et al. "A novel two-dimensional sensor with inductive spiral coils". In: *IEEE Sensors Journal* 19.13 (2019), pp. 4857–4865.

[54] Liang Wu et al. "An inductive sensor for two-dimensional displacement measurement". In: *Sensors* 20.7 (2020), p. 1819.

[55] Hang Zhou Yang et al. "A review of recent developed and applications of plastic fiber optic displacement sensors". In: *Measurement* 48 (2014), pp. 333–345.

[56] Dongdong Yu et al. "Nonlinear iterative learning control applied to an aerostatic X–Y planar motion stage". In: *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering* 226.9 (2012), pp. 1174–1182.

[57] H Yu and WJ Kim. "Controller design and implementation of six-degree-of-freedom magnetically levitated positioning system with high precision". In: *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering* 222.8 (2008), pp. 745–756.

[58] Hongxiang Yu et al. "Planar position sensor based on mono sensing electrode and hybrid-frequency excitation". In: *Sensors* 16.5 (2016), p. 691.

[59] Jian-ping Yu et al. "A planar capacitive sensor for 2D long-range displacement measurement". In: *Journal of Zhejiang University SCIENCE C* 14.4 (2013), pp. 252–257.

[60] Enzheng Zhang et al. "Laser heterodyne interferometric system with following interference units for large XY-θ planar motion measurement". In: *Optics Express* 25.12 (2017), pp. 13684–13690.

[61] Xiaodong Zhang. *Optical fiber bundle detection probe for three-dimensional blade tip clearance and demodulation method thereof*. CN111220084B. State Intellectual Property Office of the P.R.C, July 2021.

[62] Bo Zhao, Lei Wang, and Jiu-Bin Tan. "Design and realization of a three degrees of freedom displacement measurement system composed of hall sensors based on magnetic field fitting by an elliptic function". In: *Sensors* 15.9 (2015), pp. 22530–22546.

[63] Huining Zhao et al. "2D absolute position measurement based on the hybrid encoding method". In: *Review of Scientific Instruments* 94.2 (2023).

[64] L Zhou et al. "Architecture and implementation of distributed control system for MEMS-based intelligent motion surface". In: *Proceedings of the IEEE International Symposium on Industrial Electronics, 2005. ISIE 2005*. Vol. 3. IEEE. 2005, pp. 1043–1048.

[65] Lei Zhou and Jingjie Wu. "Magnetic levitation technology for precision motion systems: A review and future perspectives". In: *International Journal of Automation Technology* 16.4 (2022), pp. 386–402.