From Points to Faces: An automotive lidar-based face recognition system

M.A.R. Humblet Vertongen

From Points to Faces: An automotive lidar-based face recognition system

by

M.A.R. Humblet Vertongen

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Wednesday September 13, 2023 at 1:30 p.m.

Student number:	4580729	
Project duration:	February 2023 – September, 2023	
Faculty:	Cognitive Robotics (COR) Department of faculty 3mE, TU Delft	
Lab:	Intelligent Vehicles	
Thesis committee:	Dr. Holger Caesar,	TU Delft, Supervisor
	Dr. Xucong Zhang,	TU Delft
	Dr. Luka Peternel,	TU Delft

From Points to Faces: An automotive lidar-based face recognition system

M.A.R. Humblet Vertongen
Delft University of Technology

Abstract

Face recognition using lidar presents challenges arising from high dimensionality and data sparsity, especially at longer distances. This paper proposes a novel approach for face recognition via automotive lidar. The approach leverages a combination of deep learning and point cloud processing techniques. After identification of the facial point clouds, an alpha-shaped convex hull is employed for regional linearization, resulting in the creation of a depth image. This depth image is then fed to a convolutional neural network architecture, BasicNet, specifically trained for face recognition. The approach is evaluated on a dataset comprising 52 individuals acquired using two lidar sensors with different point densities. The individuals walked at distances ranging from 5 to 18 meters from the sensors. Results show that the approach achieves an accuracy of 63% on this challenging dataset, thereby challenging the notion that lidar sensors are privacy-preserving.

1. Introduction

Facial information technology has been investigated in a variety of security applications due to its capability of contactlessly extracting someone's private information. As society becomes more and more digital, the need for secure and effective identity verification methods has encouraged the investigation of novel solutions. Face recognition, including both verification (1:1) and identification (1:N) applications, has become an interesting research area. Numerous applications indicate the potential for an individual to be identified based on facial traits. These applications range from personalized device access to secure gate entry. Face recognition is a critical technology for security and privacy, as it can be used to authenticate individuals and track their movements.

While 2D face recognition systems have gained considerable attention, they often struggle in scenarios characterized by occlusions, shifts in illumination, or distances. Advancements in 3D face recognition show promise in addressing 2D face recognition shortcomings. The current 3D face recognition mechanisms are primarily set within the 3-meter proximity threshold [18]. 3D face recognition can be challenging due to (1) high computation power and (2) the need for high-quality and diverse data.

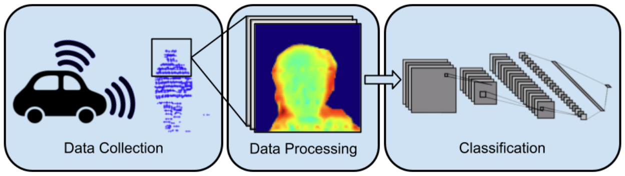


Figure 1. Overview of the research workflow. Starting with the collection of lidar data, followed by data processing, and ending with the deployment of a classification algorithm for face recognition.

The trajectory of automotive lidar, light detection and ranging, development is marked by an evolution in resolution enhancement. From the early days exemplified by Velodyne's HDL-64E automotive lidar in 2007, where a resolution of 0.33° x 1.33° was achieved [22]. More recent innovations, such as Falcon's sensor in 2022, boast a resolution surpassing 0.1° x 0.1° [13]. The domain of automotive lidar technology has witnessed exponential density progress, as can be seen in Figure 2.

The integration of such advanced sensors can raise pertinent questions concerning the privacy-related aspects of an individual. Lidar sensors are predominantly associated with applications in smart cities, intelligent mobility, and security solutions, offering insights into the understanding of the environment. Due to their low resolution, automotive lidar has been seen as a wildly privacy-preserving way to monitor people without influencing their privacy [10, 19]. Over the years, lidar sensors have become denser and more (facial) features can be observed visually.

This research outlines a novel approach for processing lidar data for face identification. To address the scarcity of 3D face recognition data, a dataset was compiled. The dataset incorporated two distinct lidar sensor types and 52 participants walking within a range of 5 to 18 meters from a vehicle containing the sensors. This research aims to answer the question of whether automotive lidar sensors can be used for person recognition applications.

The core contributions of this paper are as follows:

- Outdoor dataset creation, including two different lidar sensors and 52 distinct participants.
- Utilization of automotive lidar for face recognition.
- Utilization of an alpha-shape convex hull with linear interpolation as a processing technique.
- Development of a classifier, BasicNet, for face recognition with 52 participants.

The remainder of this paper is structured as follows: Section 2 discusses the related works, including lidar in 3D face recognition algorithms and the development of privacy-preserving lidar methods. Section 3 delves into the proposed approach. Discussing data collection, preparation, and neural network architecture for automotive lidar face recognition. Section 4 presents experimental findings, including an ablation study. Section 5 covers discussions, and Section 6 concludes the paper.

2. Related Works

2.1. Lidar-Based Face Recognition Algorithms

3D face recognition algorithms use 3D data sources, such as point clouds, depth images, and meshes, to recognize individuals. These algorithms extract distinctive features from the 3D data for the purpose of verification and identification. Efforts have also been made to explore the potential of face recognition using lidar sensors.

Ko et al. [16] used point clouds, depth maps, and RGB image data to avoid false facial verification caused by face spoofing attacks for phone identity verification. Wang et al. [23] conducted research on 3D face recognition using only point clouds acquired by a lidar sensor. They compared a single point cloud file to a 3D face model reconstructed from several point cloud files to determine the best candidate. Their dataset consisted of four individuals, all within a distance of less than 1 meter. Lim et al. [18] extended detection distances beyond the conventional limits of 3D face recognition by using a lidar sensor and amplitude image. Achieving favorable results with a dataset of two individuals spanning distances of 1 to 3 meters.

Despite the advancements, these algorithms remain restricted to close proximity (less than 3 meters) and lack the integration of automotive lidar sensors. Additionally, the datasets predominantly consist of indoor recordings, with Ko et al.'s dataset being the exception, as it includes outdoor selfie data.

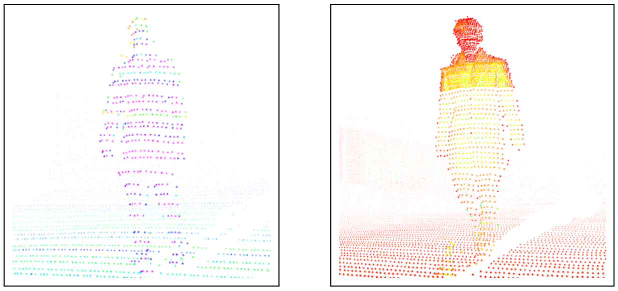


Figure 2. Standing individual at the same distance from the car for density comparison. (a) 2007 Velodyne HDL-64E (b) 2022 Innovusion Falcon with the ROI on the facial regions

2.2. Privacy-Preserving Approaches

Privacy considerations have spurred the development of privacy-preserving person-detection mechanisms. Lidar has been seen as a privacy-preserving method for navigational purposes [9] and crowd control systems [10,11,19–21]. Privacy is claimed to be preserved since objects are represented by distance values with low-resolution data points. Existing literature has yet to thoroughly explore the impact of lidar technology evolution on data density and establish comparable conclusions regarding its ongoing privacy-preserving attributes.

3. Proposed Approach

The proposed approach comprises three major components: data collection, data processing, and a face recognition algorithm. Figure 3 illustrates the data collection and the data processing pipeline.

3.1. Data Collection

Data collection was conducted using multiple sensors, including a RGB camera and two lidar sensors (the Innovusion Falcon and the Velodyne HDL-60E). The dataset consists of 52 distinct individuals with ages ranging from 20 to 40 years. The pedestrian traversal spanned a total of 140 meters and was divided into 22 walking segments. The segments, referred to as walks, have varying dimensions, directions, and proximity to the sensor-equipped vehicle. This is shown in Figure 4.

Segmentation of the total walk path was performed through manual analysis of RGB images. The initiation of each segment was determined by identifying mid-air foot motion at the starting cone and the end by foot-ground contact at the end cone. A few deviations from this criterion were observed when participants executed, e.g., turns at the end of their trajectories.

Vehicle-participant longitudinal distances range between 5 and 18 meters and extend 3.45 meters laterally on both

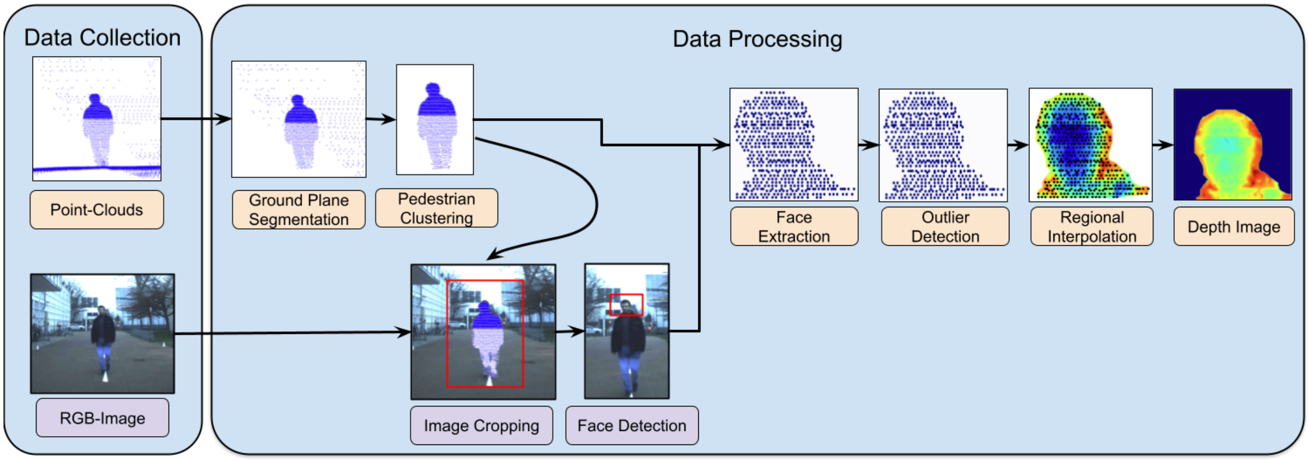


Figure 3. Data preparation pipeline, including data collection and data processing. The steps in orange are performed on lidar data, and the steps in purple are performed on RGB camera data. The camera data is solely used for determining the face region.

sides of the vehicle. Since facial features need to be visible for facial recognition, segments involving movements away from the vehicle were excluded. This refinement resulted in a subset of 16 segments, collectively spanning an approximate 122 meters.

Data collection sensors included two different lidar sensors: the Velodyne HDL-64E (2007) and the Innovusion Falcon (2022). The Velodyne sensor features angular resolutions of 0.08° to 0.35° horizontally and 0.4° vertically [22], while the Falcon sensor's angular resolution is more precise. The Falcon lidar is capable of discerning regions of interest (ROI) characterized by heightened point density. For the initial 17 participants, this ROI manifested in the lower body extremities. While the succeeding participants exhibited the ROI in facial regions. Within the ROI, the Falcon sensor offers angular resolutions of 0.09° horizontally and 0.08° vertically. Outside of the ROI, the angular resolution is 0.18° horizontally and 0.24° vertically [13]. Figure 2 shows the difference in angular resolution between the Velodyne and Falcon.

This study offers a comparative efficacy analysis of the two lidar sensors in lidar facial recognition. Nevertheless, the main focus of this study is the Falcon with the heightened density.

3.2. Data Processing

This section delineates the key steps taken during data processing. It is important to note that camera data was exclusively utilized to establish reliable ground truth areas for the face, as illustrated in Figure 3.

3.2.1 Ground Segmentation and Clustering

After filtering out the areas that were not occupied by the walking trajectory, a ground segmentation and clustering mechanism was used to find the pedestrians' point clouds.

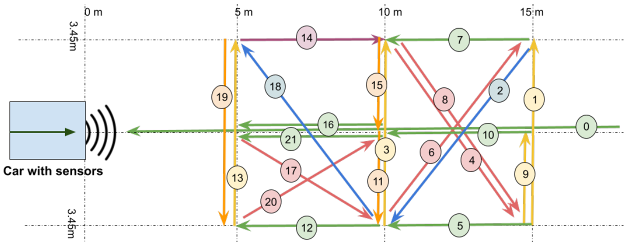


Figure 4. Complete walking trajectory with numbered walking segments: green = toward car, purple = away, yellow = left, orange = right, blue = diagonal towards car, red = diagonal away.

RANSAC (random sample consensus) was used to estimate model parameters from a set of data points [8]. RANSAC is frequently used for ground segmentation in point cloud data to distinguish between ground points (points on the ground surface) and non-ground points (points associated with objects or individuals). In this paper, the RANSAC algorithm was used with the following parameters: minimum samples of 3 and a residual threshold of 0.2. These parameters were used in all data types and scenarios. Isolating the ground plane from the point clouds can facilitate the focus on relevant objects. This includes participants, which makes pedestrian clustering easier.

The DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm was used for clustering. DBSCAN [7] is an unsupervised clustering algorithm, meaning that it does not require a ground truth label. It is based on the radius (ϵ-parameter) and the minimum number of points (min samples parameter). A point must be within the radius of another point for it to be considered a point in the cluster. This clustering algorithm is also known for being very effective in clustering datasets that may contain noise or outliers [14]. The used parameters can be found in Table 1. Under certain conditions, some of these parameters were tweaked, e.g., when intruders were too close to the pedestrian.

Longitudinal Distance from the sensor	ϵ-parameter	Minimum Samples (Falcon)	Minimum Samples (Velodyne)
$\leq$ 15 m	0.3	80	40
$>$ 15 m	0.4	50	20

Table 1. Parameters for the DBSCAN algorithm.

Even after filtering out the areas without any walking trajectory, some non-participants were still captured by lidar. In 12 walks, the clustering algorithm detected multiple possible pedestrian clusters. The selection between clusters was made manually in each case.

3.2.2 Face Detection

Given the primary focus on facial recognition, a face detection algorithm was employed to provide reliable ground truth areas.

RetinaFace [6] is a deep learning-based face detection algorithm that uses a single-shot approach. This means that it can detect faces in a single pass through the image. RetinaFace is known for its speed and high accuracy on face detection benchmarks (e.g., WIDER FACE [24]), which is why it was chosen for this paper.

RetinaFace was applied to a cropped section of the RGB camera image. This cropped section was obtained by: (1) synchronizing the two sensors (lidar and camera); (2) projecting the extracted cluster point clouds onto the image; and (3) drawing a box around the projected points. To ensure a robust area for face detection, the bounding-box was extended by an additional 30 cm in all directions.

To ensure additional reliability of the resulting face detection boxes, the following criteria were used to filter out unreliable boxes:

- The confidence interval of the face bounding-boxes from RetinaFace was at least 25%.
- Bounding-boxes that were too small (less than half the size of the previous box) or too large (twice the size of the previous box) were discarded.
- Bounding-boxes that showed no overlap with the preceding bounding-box (within one second) were also rejected.

Prior to the filtering out, the width and height of the resulting face bounding-box were doubled to provide an additional margin and to include the entire head.

The effectiveness of face detection decreased for the images of the later participants, due to decreasing illumination and reduced visibility of faces. The number of files per person in the dataset, ranged from a maximum of 850 for the first participants, to a minimum of 250 for the final participant. It is important to emphasize that the RGB image data was solely utilized for ground-truth validation within this specific context.

3.2.3 Outlier Detection with Z-Score Threshold

After preserving only the point clouds that lie within the face bounding-box area, the z-score threshold method has been applied. The z-score threshold method identifies potential outliers among the retained points.

Z-score threshold is a statistical approach that can be used for detecting outliers in point clouds [2]. In this application, the z-score was computed for each point in the face point cloud, along the three directions. The z-score measures how many standard deviations a point deviates from the mean. Points with z-scores exceeding a specified threshold are identified as outliers, and are removed from the point cloud.

The choice of the z-score threshold depends on the desired level of confidence for outlier removal. In this case, a z-score threshold of 3 was selected. This corresponds to a confidence level of 99.73%, which means that there is only a 0.27% probability that a point with a z-score of 3 or higher is not an outlier. This choice eliminates a portion of outliers within the point cloud, while retaining the majority of inlier points.

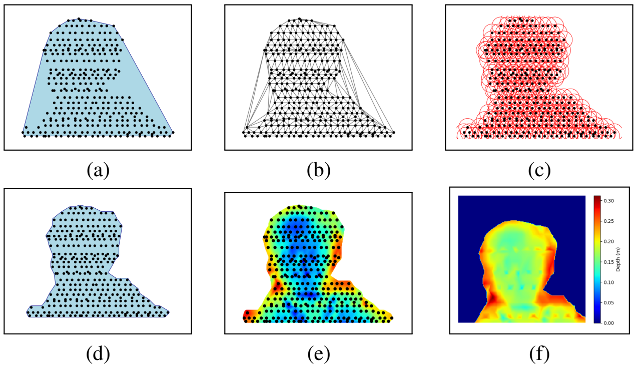
3.2.4 Alpha-Shape Convex Hull and Linear Interpolation

Following the data processing process, the next step is to determine the interpolation region. This was done by using the alpha-shape convex hull (ASCH) method [3]. The ASCH is able to create more accurate and detailed boundaries around point clouds than the traditional convex hull method (see Figure 5a). The alpha shape is a geometric construct defined by a set of points in a plane. Each point is connected to its neighbors if they fall within a predetermined distance of each other. The parameter that influences the distance between this connection is denoted as the alpha parameter. The radius r is calculated as a function of this alpha parameter: $r = 1/\alpha$. The alpha parameter governs the presence of voids within the polygon. A smaller alpha parameter results in fewer voids, yielding a denser shape. Whereas, A larger alpha parameter results in a sparser shape. In this paper, the alpha parameter is chosen based on the number of face point clouds present. This choice is made because the number of face point clouds depends on the distance from the sensor and on the specifications of the sensor. The alpha values have been determined through manual tuning and are displayed in Table 2.

Points	$[0, 100[$	$[100, 200[$	$[200, 400[$	$[400, \infty[$
α-value	0.05	0.1	0.2	0.25

Table 2. Alpha-values based on the number of point clouds

Computing the ASCH for a given set of points involves employing the Delaunay triangulation [5]. The Delaunay triangulation (Figure 5b) optimally divides the points into triangles, minimizing the intersections among them. A triangle area is kept when the points making the triangle are within each other's radius (Figure 5c). The resulting area (Figure 5d) is subsequently used for linear interpolation between the points (Figure 5e).



(a) (b) (c)

(d) (e) (f)

Figure 5. Steps taken from the initial extracted face point clouds to the depth image (a) Convex Hull (b) Delaunay Triangulation (c) Radius around points (d) ASCH (e) ASCH polygon with interpolation (f) Final depth image.

3.2.5 Depth Image Generation

The remaining points are rasterized with pixel dimensions of 0.005 by 0.005 m. This choice was made arbitrarily through manual tuning. Additionally, the maximum height of a face is cropped at 30 cm. This value was chosen based on the knowledge that the average face size of a fully grown man is 19cm [17].

Each pixel in the depth image corresponds to a value in meters. Pixels initially containing a NaN value (representing the background) were assigned a value of 0 m. Pixels containing an initial depth value of x m received the following adjustment: $x+0.1-\delta$ where δ represents the minimum value. The addition of 0.1m helps distinguish these values from the background values.

This process results in a final 224 by 224 depth image, visible in Figure 5f, which will be used as input for the neural network.

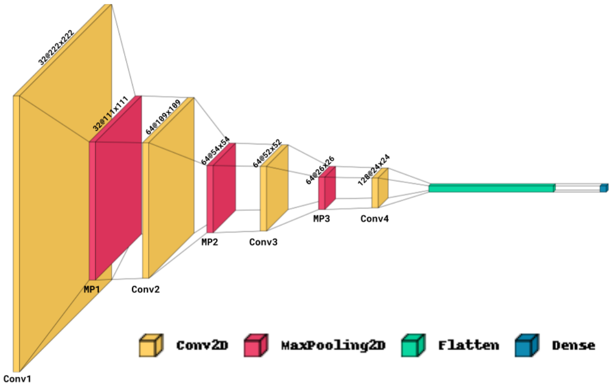


Figure 6. BasicNet architecture. With the number of channels and size per layer.

3.3. Neural Network Architecture

This section introduces the neural network architectures developed for lidar-based face recognition, treating it as a 52-class classification task. The network, named BasicNet, adopts a simple structure with a few layers. Figure 6 illustrates the network architecture, involving convolutional layers for feature extraction and fully connected layers for classification.

The design choices for BasicNet include four convolutional layers with 3x3 convolutional filters to retain spatial information using the ReLU activation function [1]. The number of filters per layer is 32, 63, 63, and 124, respectively. Three max-pooling layers with a 2x2 pool size and a 2x2 stride are placed between the convolutional layers. Additionally, there is a fully connected layer for linking to the output and a softmax output layer with 52 classes for image classification. The input data for the network is a one-channel, 224x224, non-normalized depth image. The output corresponds to one of the 52 classes.

4. Experiments and Results

In this section, the experimental setup and results are presented. Firstly, an ablation study for the BasicNet was performed on a restricted dataset. Then, tests were performed on a larger dataset, including the Velodyne and Falcon datasets.

4.1. Ablation Study of Network Architecture

An ablation study is a research method in which components of a system are removed or modified to assess their relative impact on the system's overall performance. In this paper, two types of ablation tests were conducted. First, the impact of the network's depth was investigated. Second, the workings of individual elements were examined.

The following setup was used during the totality of the ablation tests. The ablation experiments were performed on a smaller dataset containing only walks 16 and 21. These walks have the same walking paths but were conducted at different times. One walk (walk 16) was used as training data and the other (walk 21) as testing data, with a random 80/20 train-validation split. Only the interpolated Falcon data was utilized for the ablation experiments. The optimizer used is Adam with a learning rate of 0.0001 [15]. Categorical cross-entropy was employed as the loss function. An early stopping mechanism with a patience of 5 was implemented, which means that training stops when the accuracy has been lower for 5 consecutive epochs. The weights of the model with the best accuracy are saved.

4.1.1 Depth Analysis

Several ablation experiments were conducted on the BasicNet model to understand how its depth affects test accuracy. To start, a single convolutional layer with a filter size of 32 was used. A max-pooling layer with a kernel size of 2x2 and a stride of 2x2 was then added, along with an additional convolutional layer with a filter size of 64. This procedure was repeated three times: once with a convolutional layer with a filter size of 64 and twice with a filter size of 128. The structure of the network up to seven layers is shown in Figure 6. Table 3 depicts the relationship between the model's depth and its test accuracy for up to nine layers. Based on this test, the BasicNet model with three max-pooling and four convolution blocks achieved the highest accuracy.

Layers	1	3	5	7	9
Test Accuracy	75 %	76 %	79 %	**84 %**	82 %

Table 3. BasicNet's depth analysis results. BasicNet is the 7-layered network that achieved an accuracy of 84%.

4.1.2 Individual Network Element Analysis

Next, ablation experiments were conducted on individual network elements, using the same experimental setup as during the depth analysis. The activation function was changed first, followed by the removal and modifications of max-pooling and convolutional layers. These experiments yielded a total of 10 different ablation tests.

The results, shown in Table 4, indicate that the activation function has a significant influence on the test accuracy. Additionally, single max-pooling layers showed a greater influence on accuracy compared to single convolutional layers.

4.2. Model Deployment

After the ablation study, the model was deployed on the entire dataset. The BasicNet network was trained and tested on interpolated and non-interpolated Velodyne and Falcon data. All experiments were conducted under the same conditions: and Adam optimizer with a learning rate of 0.0001, a categorical cross-entropy loss function, and early stopping with a patience of 5 epochs. The training and testing were executed on an ARM-based system, the Apple M2 chip.

Description	**Accuracy**
BasicNet	*84 %*
Activation function: ReLU → ELU [4]	78 %
Activation function: ReLU → Tanh	10 %
Activation function: ReLU → Sigmoid	2 %
Max pooling: Remove MP1	77 %
Max pooling: Remove MP2	80 %
Max pooling: Remove MP3	78 %
Max pooling: MP3 → along-channel max pool	82 %
Convolution: Remove Conv2, Conv3 or Conv4	81 %
Convolution: Filter size Conv3, 64 → 128	82 %
Convolution: Filter size Conv4, 128 → 64	81 %

Table 4. BasicNet's individual network element analysis results. The test accuracy refers to the accuracy of the test set. → stands for ”replaced with”.

The dataset contains 52 different individuals. The total number of training files is 21.5k for the Falcon sensor and 21k for the Velodyne sensor. The total number of test files is around 7K for both sensors, with 4K straight walks and 3K side walks.

The BasicNet network was trained and tested on three different scenarios: straight walking paths, side walking paths, and combined straight and side walking paths. The training-test split was done walk-wise. This means that the same walk (walk 00) was selected for straight testing purposes for all individuals. In the same way, two smaller side walks were set aside (walk 11 and walk 15) for side testing. The training-validation split was done file-wise and therefore unique to each sensor with a random 80/20 train-validation split.

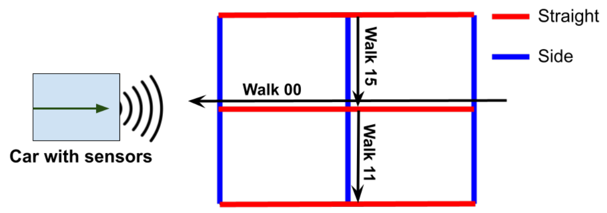


Figure 7. Straight and side walking path directions and the test walking segments.

4.2.1 Training on Straight Walking Paths

The network was trained on only straight walking paths and tested on straight and side walking paths separately. For a random classifier, the accuracy rate is 1 divided by the number of classes. In this case, a random classifier would have an accuracy of $\frac{1}{52} = 1.9\%$.

Table 5 presents the test results for BasicNet. The test accuracy is calculated by dividing the number of correct predictions by the total number of predictions. The macro-averaged F1-score (or macro F1-score) is computed using an unweighted mean of all the per-class F1 scores. The macro F1-score treats all classes equally.

The network achieved the highest test accuracy and macro F1-score on the interpolated Falcon data. The model was also tested on its ability to predict side data. The accuracy rate is lower on the side test compared to the straight test. Notably, the interpolated and non-interpolated versions of a data sensor have similar accuracy on the side test.

Training on straight walking paths

	Straight Test		*Side Test*	
	Accuracy	Macro F1	Accuracy	Macro F1
Falcon	85 %	85 %	15 %	12 %
Falcon*	69 %	68 %	14 %	11 %
Velodyne	33 %	33 %	6 %	5 %
Velodyne*	19 %	18 %	5 %	4 %

Table 5. Network results for BasicNet trained on straight walking paths and tested on straight and side walking paths separately. The * depicts non-interpolated data.

4.2.2 Training on Side Walking Paths

This network was trained on side walking paths and separately tested on straight and side walking paths. Performance results are shown in Table 6. BasicNet trained on side walking paths did not outperform the network trained on straight walking paths. The interpolated Falcon data achieved the best performance on both test sets. Notably, the interpolation did not improve the test scores for the straight test. An interesting observation is that the Velodyne trained on straight and side walking paths have very similar results when tested on the side test.

The lower performance results of BasicNet trained on side walking paths in comparison to training on straight walking paths can have several reasons. Additionally, the Falcon sensor uses a visible horizontal scanning pattern in the ROI, which results in a shift in the face area, as illustrated in Figure 8. The amplitude of the horizontal shift is related to the horizontal speed, and therefore more visible on faces in side walking paths. Nevertheless, the Velodyne also had more difficulty identifying side faces. Therefore, side-face features may be less prominent than front-face features, making it more difficult for the classifier to distinguish between the participants.

Training on side walking paths

	Straight Test		*Side Test*	
	Accuracy	Macro F1	Accuracy	Macro F1
Falcon	11 %	10 %	35 %	33 %
Falcon*	11 %	10 %	22 %	20 %
Velodyne	4 %	4 %	7 %	6 %
Velodyne*	5 %	4 %	6 %	4 %

Table 6. Network results for BasicNet trained on side walking paths and tested on straight and side walking paths separately. The * depicts non-interpolated data.

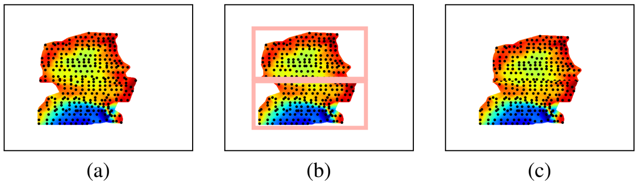

(a) (b) (c)

Figure 8. Manually reconstructed side face using interpolated data. (a) Original side face (b) Visually detected scanning regions (c) Manually recreated side face.

4.2.3 Training on Straight and Side Walking Paths

Finally, the network was trained and tested on straight and side walking paths collectively. As mentioned previously, a random classifier would achieve an accuracy score of approximately 1.9%. Therefore, all results in Table 7 have performed better than a random classifier. The findings of this research in all scenarios indicate that the interpolated Falcon model stands out as the best choice for face recognition across all perspectives.

Training on straight & side walking paths

	Straight & Side Test	
	Accuracy	Macro F1
Falcon	63 %	63 %
Falcon*	47 %	47 %
Velodyne	21 %	21 %
Velodyne*	13 %	13 %

Table 7. Network results for BasicNet trained and tested on straight and side walking paths collectively. The * depicts non-interpolated data.

As stated in Section 3.1, the first 17 individuals do not have the Falcon's region of interest (ROI) in their facial area. This reduction in point-cloud density resulted in a lower individual F1-score. This held true for both the interpolated and non-interpolated Falcon data. BasicNet using interpolated Falcon data achieved an average F1-score of 62% for the first 17 individuals, compared to 74% on the other individuals. For the non-interpolated Falcon data, an average F1-score of 35% was achieved for the first 17 individuals, and 51% on the other individuals. What has been noticed is that the 17 first individuals are more susceptible to being misidentified amongst each other, especially in the non-interpolated Falcon data. Also, individuals with longer hair tend to achieve higher F1-scores.

Both Falcon data types, interpolated and non-interpolated, had the lowest accuracy towards the same participant, with an accuracy rate of 31% and 18% respectively. This participant was among the first 17 participants and has generic facial features. It was also noticed that the non-interpolated Falcon data needed more epochs (13 compared to 8 for interpolated) to achieve the optimal solution. Concerning Velodyne, the individual with the cowboy hat was the most noticed of all, with an F1-score of 50%. This is extraordinary since the average accuracy and macro F1-score is of 21%.

It should be noted that BasicNet is a very simple model and was used to illustrate that even a simple model is enough to identify an individual. It should also be noted that interpolation increased generalizability and most accuracy rates.

The same experiments were also conducted on ResNet18 and ResNet50 architectures. However, the training curves showed unusual behavior with abrupt changes. Hence, no conclusion was drawn using this data. The data for additional tests on ResNet architectures conducted is attached in the appendix.

5. Discussion

The experimental findings of this study suggest that lidar sensors can be used for face recognition. Nevertheless, it is essential to acknowledge and address certain limitations. These limitations can present opportunities for future research.

Firstly, the **dependency on head related features.** The use of hair and head accessory information on the depth images, introduces a potential dependency on these features beyond the face. This could limit the generalizability of the findings to individuals depending on their hair or headwear. Notably, consistent positive results were observed in all training scenarios and networks for the individual wearing a cowboy hat and the better results with long haired individuals.

The **treatment of Velodyne data**. While the approach in this study was able to enable facial detection using automotive lidar, there may be room for improvement in Velodyne's effectiveness in recognizing individuals. The Velodyne data was treated in the same way as the Falcon data. However, some ablation tests specifically on Velodyne data could have created a Velodyne-based system with better accuracy.

Falcon's in ROI **scanning pattern**: A noticeable horizontal shift in facial recognition was observed, which becomes more pronounced during side walks, particularly around the middle of the walk when participants have higher velocities. This discrepancy likely arises from the horizontal scanning pattern of the Falcon. Consequently, this shift may have adversely affected the results.

Compared to the straight test data, the walks in the **side test data** have less similar walking paths in the training data. Walks at the same distance were present in the training data, but they were walked in the opposite direction (from left to right instead of right to left). It should also be noted that exactly the same walking path was not present in the test set of straight walking paths either.

The **exclusion of diagonal walking paths**: When the model was trained with the inclusion of diagonal walk data, it yielded unsatisfactory results. This could be attributed to its underrepresentation compared to other walking directions (total of 2 walks), in addition to the complexity introduced by the horizontal scanning pattern. Therefore, this data was not included in these experiments.

It is an **incomplete lidar-based system**. The current pipeline does not constitute a complete lidar-based system. By incorporating components such as face detection on range images, a fully lidar-based recognition system could be established without any reliance on RGB images. Consequently, it would operate effectively even in darker scenarios. This expansion holds promise for refining the overall system's capabilities since the face detector used in this study, RetinaFace, provided fewer detections in darker environments.

Privacy considerations: Given the non-negligible accuracy achieved, privacy concerns are difficult to dismiss. Perfect privacy with lidar will be difficult to achieve, but there are a number of potential approaches that could be taken. One potential approach is to impose a maximum point density per distance for future lidar sensors. Inspired by RGB image blurring, adding an average box of point clouds where a face is detected could be considered.

6. Conclusion

This paper presents a innovative method for automotive lidar-based face recognition. An outdoor dataset was created consisting of 52 individuals walking at distances ranging from 5 to 18 meters. Various sensors, including two automotive lidars with different densities, were employed. Data processing techniques such as clustering, outlier detection, and alpha-shape convex hull with interpolation were applied.

Using the proposed approach, an identification accuracy of 64% was achieved, which rose to 84% for straight walking paths. On the lower-density lidar sensor, a respectable 33% accuracy was achieved, considering that a random classifier would achieve only 2%. This indicates that even a simple architecture like BasicNet can successfully identify individuals, raising potential privacy concerns.

The increased data density has made automotive lidar a more practical choice for identification purposes, eliminating the need for custom sensors. Furthermore, the findings underscore the significance of interpolation in enhancing accuracy by up to 16% in specific scenarios and promoting greater generalization across density types. This represents an advancement in face recognition using automotive lidar sensors.

7. Future Work

Future work could concentrate on enhancing the performance of the proposed method with limited head features, focusing exclusively on face data. Furthermore, exploring the horizontal pattern within Falcon's region of interest (ROI) presents another promising avenue for investigation.

References

[1] Abien Fred Agarap. Deep Learning using Rectified Linear Units (ReLU), Feb. 2019. arXiv:1803.08375 [cs, stat]. 5

[2] Vaibhav Aggarwal, Vaibhav Gupta, Prayag Singh, Kiran Sharma, and Neetu Sharma. Detection of Spatial Outlier by Using Improved Z-Score Test. In *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, pages 788–790, Apr. 2019. 4

[3] Saeed Asaeedi, Farzad Didehvar, and Ali Mohades. Alpha-Concave hull, a generalization of convex hull. *Theoretical Computer Science*, 702:48–59, Nov. 2017. 4

[4] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs), Feb. 2016. arXiv:1511.07289 [cs]. 6

[5] Boris Delaunay. Sur la sphere vide. *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk*, 7(793-800):1–2, 1934. 5

[6] Jiankang Deng, Jia Guo, Yuxiang Zhou, Jinke Yu, Irene Kotsia, and Stefanos Zafeiriou. RetinaFace: Single-stage Dense Face Localisation in the Wild, May 2019. arXiv:1905.00641 [cs]. 4

[7] Martin Ester, Hans-Peter Kriegel, and Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. 1996. 3

[8] Martin A. Fischler and Robert C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Apphcatlons to Image Analysis and Automated Cartography. SRI International, 1981. J. D. Foley Editor. 3

[9] Ting Gu, Kiho Lim, Gyu Ho Choi, and Xiwei Wang. A Lidar Information-based Privacy-Preserving Authentication Scheme Using Elliptic Curve Cryptosystem in VANETs. In *2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC)*, pages 525–526, Jan. 2022. ISSN: 2331-9860. 2

[10] Andrei Günter, Stephan Böker, Matthias König, and Martin Hoffmann. Privacy-preserving People Detection Enabled by Solid State LiDAR. In *2020 16th International Conference on Intelligent Environments (IE)*, pages 1–4, July 2020. ISSN: 2472-7571. 1, 2

[11] Mahmudul Hasan, Junichi Hanawa, Riku Goto, Hisato Fukuda, Yoshinori Kuno, and Yoshinori Kobayashi. Tracking People Using Ankle-Level 2D LiDAR for Gait Analysis. In Tareq Ahram, editor, *Advances in Artificial Intelligence, Software and Systems Engineering*, volume vol 1213. of *Advances in Intelligent Systems and Computing*, pages 40–46. Springer International Publishing, Cham, June 2021. 2

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition, Dec. 2015. arXiv:1512.03385 [cs]. 13

[13] Innovusion, Inc. Falcon Kinetic LiDAR System User Manual V1.0. Manual UM-EN-P-V1.0-20220412, Innovusion, Apr. 2022. 1, 3

[14] Kamran Khan, Saif Ur Rehman, Kamran Aziz, Simon Fong, and S. Sarasvady. DBSCAN: Past, present and future. In *The Fifth International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2014)*, pages 232–238, Feb. 2014. 4

[15] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, Jan. 2017. arXiv:1412.6980 [cs]. 6

[16] Kyoungmin Ko, Hyunmin Gwak, Nalinh Thoummala, Hyun Kwon, and SungHwan Kim. SqueezeFace: Integrative Face Recognition Methods with LiDAR Sensors. *Journal of Sensors*, 2021:e4312245, Sept. 2021. Publisher: Hindawi. 2

[17] Wonsup Lee, Jangwoon Park, Jeong Rim Jeong, Eunjin Jeon, Hee-Eun Kim, Seikwon Park, and Heecheon You. *Analysis of the Facial Anthropometric Data of Korean Pilots for Oxygen Mask Design*, volume 56. Oct. 2012. Journal Abbreviation: Proceedings of the Human Factors and Ergonomics Society Annual Meeting Publication Title: Proceedings of the Human Factors and Ergonomics Society Annual Meeting. 5

[18] Yoon-Seop Lim, Sung-Hyun Lee, Sung-Jin Cheong, and Yong-Hwa Park. A long-distance 3D face recognition architecture utilizing MEMS-based region-scanning LiDAR. In *MOEMS and Miniaturized Systems XXII*, volume 12434, pages 87–91. SPIE, Mar. 2023. 1, 2

[19] Masakazu Ohno, Riki Ukyo, Tatsuya Amano, Hamada Rizk, and Hirozumi Yamaguchi. Privacy-preserving Pedestrian Tracking using Distributed 3D LiDARs. Jan. 2023. 1, 2

[20] Bruno Rodrigues, Lukas Müller, Eder J. Scheid, Muriel F. Franco, Christian Killer, and Burkhard Stiller. LaFlector: a Privacy-preserving LiDAR-based Approach for Accurate Indoor Tracking. In *2021 IEEE 46th Conference on Local Computer Networks (LCN)*, pages 367–370, Oct. 2021. ISSN: 0742-1303. 2

[21] Onur N. Tepencelik, Wenchuan Wei, Leanne Chukoskie, Pamela C. Cosman, and Sujit Dey. Body and Head Orientation Estimation with Privacy Preserving LiDAR Sensors. In *2021 29th European Signal Processing Conference (EUSIPCO)*, pages 766–770, Aug. 2021. ISSN: 2076-1465. 2

[22] Velodyne LiDAR, Inc. HDL-64E Technical Documentation. Technical report, Velodyne, 2017. 1, 3

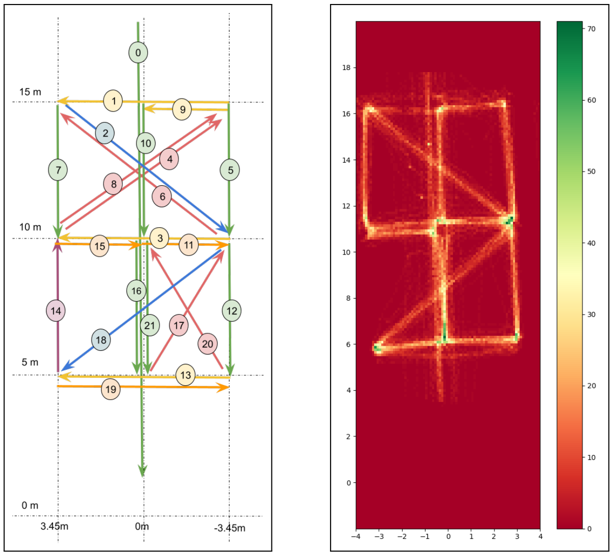
[23] Cheng-Wei Wang and Chao-Chung Peng. 3D Face Point Cloud Reconstruction and Recognition Using Depth Sensor. *Sensors*, 21(8):2587, Jan. 2021. Number: 8 Publisher: Multidisciplinary Digital Publishing Institute. 2

[24] Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. WIDER FACE: A Face Detection Benchmark, Nov. 2015. arXiv:1511.06523 [cs]. 4

A. Supplementary Material: Dataset Collection

A.1. Walking Segments

As described in Section 3.1, the entire walking trajectory was divided into 22 segments, labeled from 0 to 21. Figure 9a shows the complete trajectory, with each direction marked by a distinct color. The heatmap in Figure 9b shows the detected faces in the segments used in this paper. It is important to note that the minimum confidence threshold used for face detection was 0.25.



(a)

(b)

Figure 9. Visualization of walking paths and face detection heat maps. (a) Complete walking path segments: green = toward car, purple = away, yellow = left, orange = right, blue = diagonal towards car, red = diagonal away. (b) A heat map illustrating the face detection results for all participants across the entire walking area

A.2. Participants

To provide complete transparency regarding potential biases in the dataset, a few additional insights into the participants are presented. A total of 52 participants took part in the study. The average duration of the total walking trajectory per participant was 2.13 minutes, with the longest walk being 1.68 minutes (participant 24) and the shortest at 2.57 minutes (participant 27). Some additional observations about the participants: (1) Some walks contain errors. A small portion of participants deviated from the designated path, walking in the wrong direction or not walking in a straight line. This occurred in a total of 3% of the walks. (2) The gender distribution of participants was 21.2% female and 78.8% male. (3) Another additional statistic involves the occurrence of non-participants within the sensor's field of vision, which was recorded in 5.1% of the walks. (4) It should be noted that three individuals were spotted wearing headgear, including a cowboy hat.

B. Supplementary Material: Data Processing

B.1. Face Detection

Not all participants were equally detected, as shown in Figure 10. There is a noticeable decline in the number of files per participant. Possible reasons for this variation include differences in walking duration and lighting conditions. Faster walkers spend less time on the path, resulting in fewer detected faces. Additionally, the effectiveness of the RetinaFace face detection algorithm may be influenced by ambient lighting. Data collection took place in February in the northern hemisphere, and darker lighting conditions prevailed around 5:15 p.m. Data collection continued until approximately 6:30 p.m.

As shown in Figure 11, the average face detection confidence interval does not decrease as significantly as the number of files per participant. There is an abrupt decrease in face detection confidence for participant 25. Notably, participant 25 was wearing a high-necked jacket and a scarf. This distinction is considered the cause of the decrease in confidence level; however, the exact cause of this decrease remains unexplained.

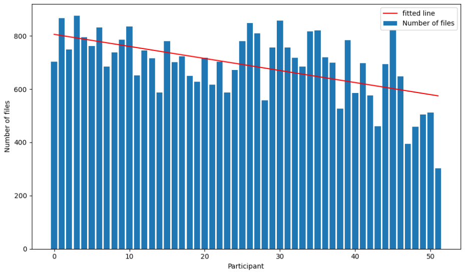


Figure 10. Number of RGB-image files with a detected face (confidence > 0.25) per participant.

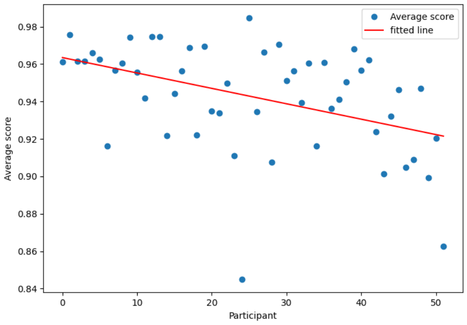


Figure 11. Average RetinaFace face detection confidence per participant.

B.2. Z-Score Threshold

To mitigate biases in the face point clouds, a z-score threshold of 3 was used. The z-score formula is given by:

$$z = \frac{x - \mu}{\sigma}$$

where μ is the mean and σ is the standard deviation.

The z-score threshold of 3 was chosen after experimenting with different values. This threshold was found to be effective in removing outliers and reducing bias in the face point clouds.

B.3. ASCH: Alpha-Value

As discussed in Section 3, the alpha value has an influence on the number of voids in the ASCH. A high alpha-value results in more voids than a lower alpha-value, as can be seen in Figure 12. In this case, a polygon without any voids is desired while preferably keeping the complete head shape, such as the neck, visible.

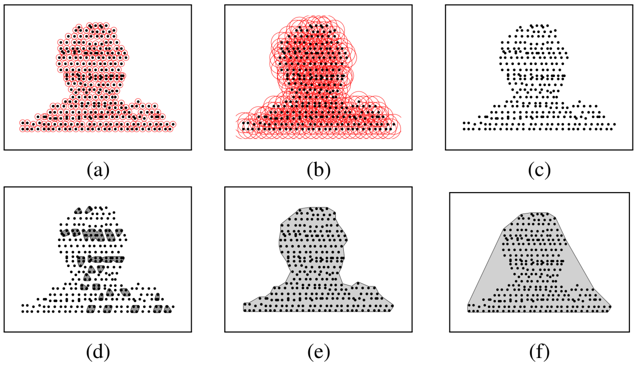

(a) (b) (c)
(d) (e) (f)

Figure 12. Illustration of the alpha influence on the number of voids. The top row, illustrating the radius with different alpha-values for (a) alpha of 0.5 (b) alpha of 0.15 (c) alpha of 0.001, is too large to be visible on the figure. The bottom row illustrates the ASCH (d) alpha of 0.5 (e) alpha of 0.15 (f) alpha of 0.001, acting as a regular convex hull.

B.4. Side Faces

As discussed in Section 3, several steps were taken to process the data before making it input-ready. After the final face points were selected and the outliers were removed, the alpha-shape convex hull (ASCH) method was used to create a polygon in which a linear interpolation was performed. These steps are demonstrated on a side image in Figure 13.

Through these processing steps, a visible region scanning pattern became apparent. Shifts were observed in the point clouds due to the scanning pattern in the high-density point cloud region of the Falcon. Figure 13c provides an example.

Some small modifications were attempted, such as an ICP variation and K-D trees, but none were successful. Therefore, the face in Figure 8 was manually reconstructed in order to provide an example.

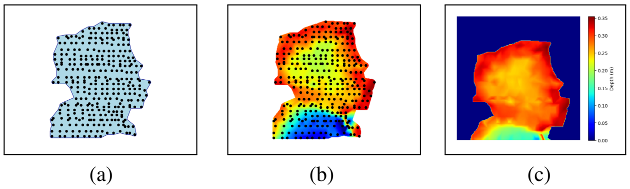


Figure 13. Creation of final depth image for Side Face. (a) ASCH Polygon (b) Region with interpolation (c) Final side face depth image

C. Supplementary Material: Network Architecture and Results

C.1. Data-Split Numbers

The following table summarizes the number of lidar files in each data split scenario. It should be noted that the interpolated and non-interpolated alternatives are based on the same files.

Sensor	Scenario		Count
Falcon	Straight	Train	10,098
		Validation	2,524
		Test	4,499
	Side	Train	7,159
		Validation	1,789
		Test	2,720
	All	Train	17,256
		Validation	4,314
		Test	7,219
Velodyne	Straight	Train	9,844
		Validation	2,460
		Test	4,185
	Side	Train	7,000
		Validation	1,749
		Test	2,609
	All	Train	16,843
		Validation	4,210
		Test	6,794

Table 8. File counts per training scenario. The ”all” scenario refers to side and straight walking patterns.

C.2. Accuracy Calculations

In subsection 4.2, accuracy and macro F1-score were discussed. This section will look more closely at the meaning of the terms and their equations.

$$\text{accuracy} = \frac{\text{correct predictions}}{\text{total predictions}}$$

$$\text{F1}_i = \frac{2 * \text{precision}_i * \text{recall}_i}{\text{precision}_i + \text{recall}_i}$$

where i is the class index.

$$\text{precision}_i = \frac{\text{true positives}_i}{\text{true positives}_i + \text{false positives}_i}$$

$$\text{recall}_i = \frac{\text{true positives}_i}{\text{true positives}_i + \text{false negatives}_i}$$

There are two averages that are taken into account: the macro average of the F1 score and the weighted average. Macro F1 is the unweighted average of the F1 scores for each class. The paper showed the macro average since it provides an equal representation for all classes.

$$\text{Macro F1} = \frac{1}{n}\sum_{i=1}^{n} \text{F1}_i$$

where n is the number of classes.

Weighted F1 is the weighted average of the F1 scores for each class. It is a good measure of accuracy when the classes are not equally important. The weights are given by the number of samples in each class.

$$\text{Weighted F1} = \frac{\sum_{i=1}^{n} w_i \text{F1}_i}{\sum_{i=1}^{n} w_i}$$

where w_i is the number of samples in class i.

C.3. ResNet

In addition to BasicNet, two ResNet [12] network architectures were investigated. ResNet18 and ResNet50 are both convolutional neural networks that are based on the residual learning framework. The main difference between ResNet18 and ResNet50 is the number of layers. ResNet18 has 18 deep layers, while ResNet50 has 50 deep layers. This means that ResNet50 has more parameters and is more computationally expensive to train. A global average pooling and dense layer were added to the backbones of ResNet18 and ResNet50 to make them suitable for multi-class classification. The architectures of ResNet18 and ResNet50 are shown in Figure 14.

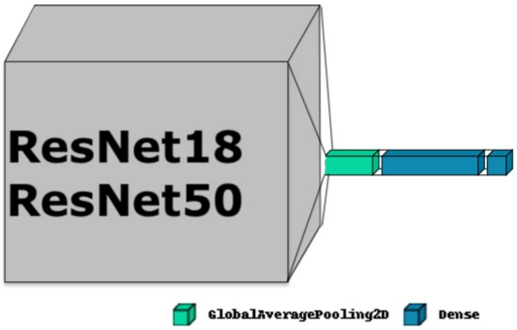


Figure 14. ResNet18 and ResNet50 architectures adapted to classification

The ResNets were trained on the interpolated Falcon data. The training and testing conditions were the same as with BasicNet: Adam optimizer, learning rate of 0.0001, categorical cross-entropy loss function, and early stopping with a patience of 5. The training and testing were on an ARM-based system, the Apple M2 chip.

Regarding the ResNet results, it can be said that they outperformed BasicNet in most scenarios. When trained on the straight data, BasicNet performed as well as ResNet18 on the straight test and similarly on the side test (Table 9). It should be noted that ResNet also performed less when trained on the side walking paths (Table 10). On the complete data, ResNet18 achieved a macro F1 score of 78% (Table 11). ResNet18 outperformed the other architectures, including ResNet50.

Nevertheless, the training process for ResNet networks was not as expected. The learning curves showed peaks, which is not characteristic of a healthy learning process. The learning curves for ResNet18 and ResNet50 can be seen in Figures 17 and 18. In comparison to the learning curves of BasicNet on interpolated Velodyne and Falcon data, the peaks are not present (16 and 15). This could be due to the architecture being too complex for the task. A smoothing factor of 0.6 has been added to the learning curves to have a better grasp of the mean tendency of the curve. Even with the smoothing factor, ResNet18 and ResNet50 do not perform as stable as BasicNet.

Training on side walking paths

	Straight Test		*Side Test*	
	Accuracy	Macro F1	Accuracy	Macro F1
ResNet18	85 %	85 %	14 %	13 %
ResNet50	81 %	80 %	10 %	10 %

Table 9. Network results for ResNet trained on interpolated Falcon straight walking paths and tested on interpolated Falcon straight and side walking paths separately.

Training on side walking paths

	Straight Test		*Side Test*	
	Accuracy	Macro F1	Accuracy	Macro F1
ResNet18	16 %	13 %	42 %	39 %
ResNet50	13 %	9 %	35 %	32 %

Table 10. Network results for ResNet trained on interpolated Falcon side walking paths and tested on interpolated Falcon straight and side walking paths separately.

Training on straight & side walking paths

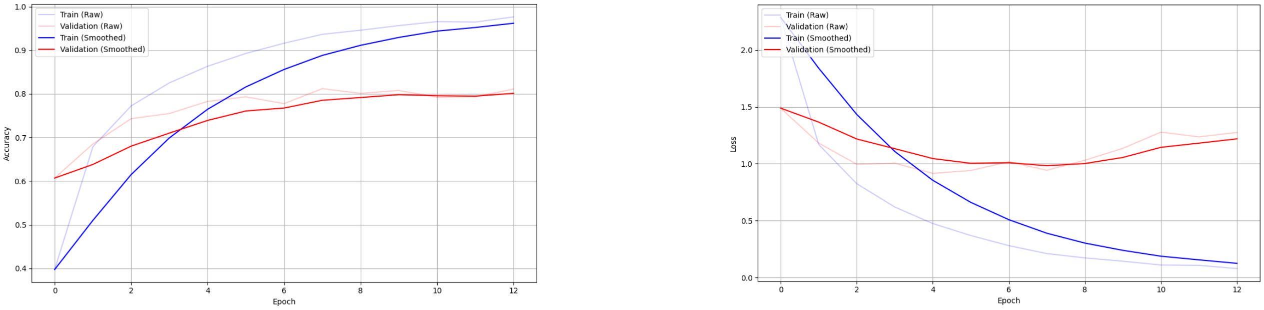
	Straight & Side Test	
	Accuracy	Macro F1
ResNet18	77 %	78 %
ResNet50	69 %	69 %

Table 11. Network results for ResNet trained and tested on interpolated Falcon straight and side walking paths collectively.

C.4. Confusion Matrix Insights

As mentioned in Section 4, the addition of interpolation has made it less likely to confuse individuals with high and low density in the falcon data. When examining the confusion matrices in Figure 19b, it can be observed that in the non-interpolated data, the first 16 individuals are frequently misidentified among themselves. In Figure 19a it can be noticed that when individuals are misidentified, they are distributed more evenly across density categories, although still slightly skewed towards higher or lower density.

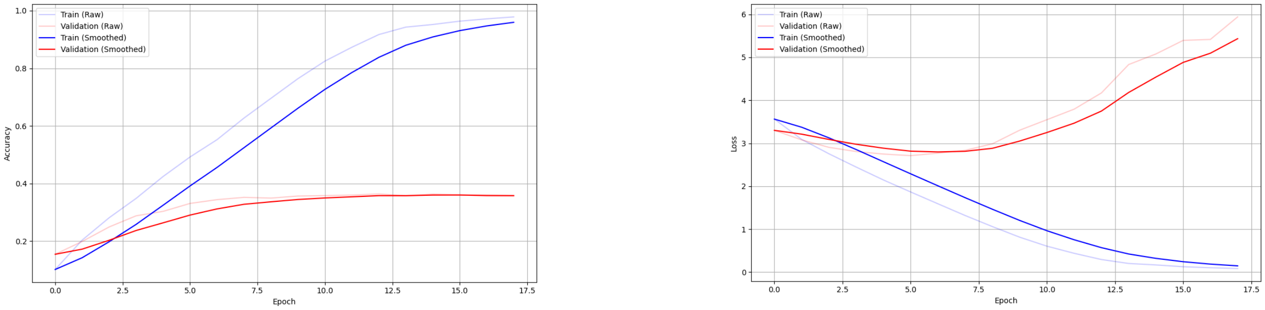
Upon comparing Figure 20 and Figure 19a, it becomes apparent that there are more misidentified individuals in the Velodyne data compared to the Falcon data.



(a) Accuracy over Epoch

(b) Loss over Epoch

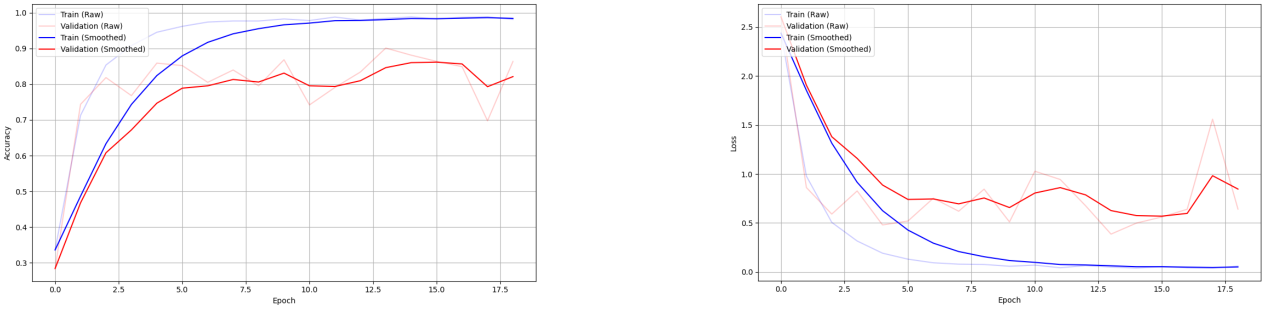
Figure 15. BasicNet learning curves, on interpolated Falcon data in the all walks scenario with blue training and red validation.



(a) Accuracy over Epoch

(b) Loss over Epoch

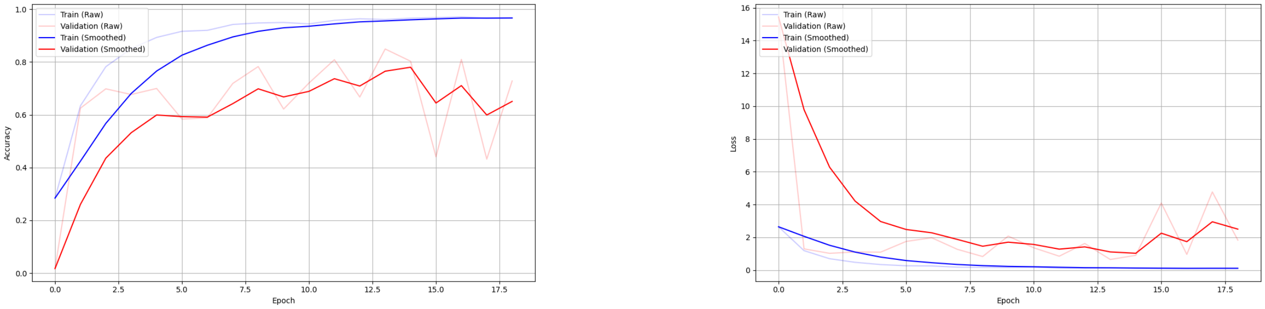
Figure 16. BasicNet learning curves on interpolated Velodyne data in the all walks scenario with blue training and red validation.



(a) Accuracy over Epoch

(b) Loss over Epoch

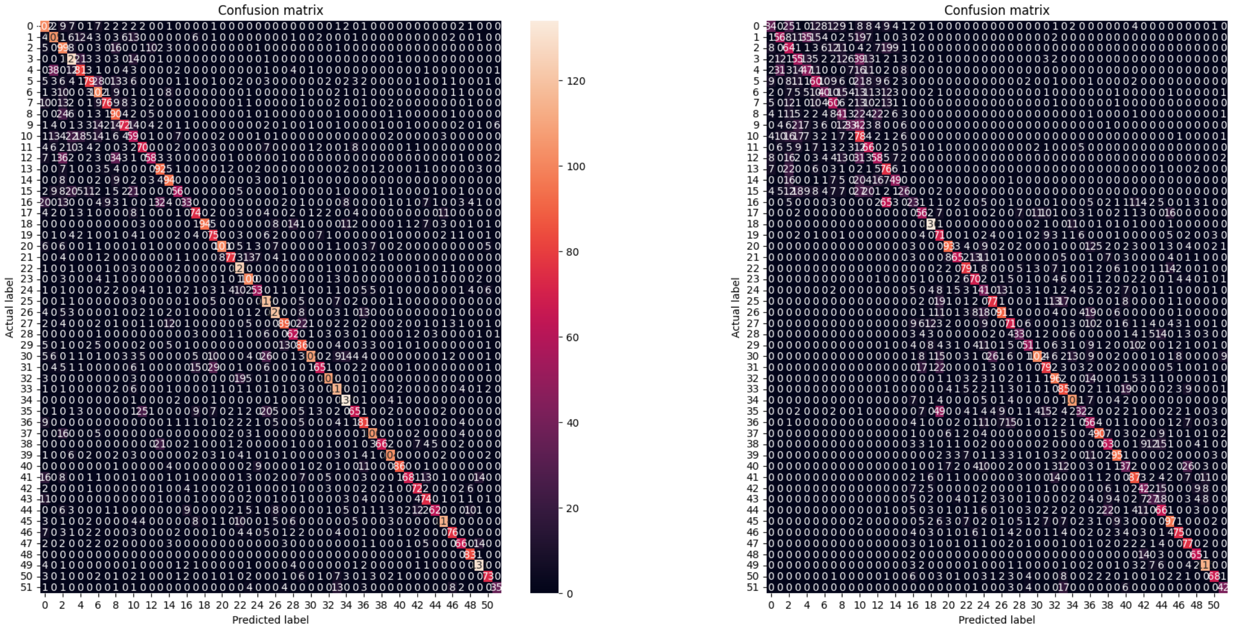
Figure 17. ResNet18 learning curves, on interpolated Falcon data in the all walks scenario with blue training and red validation.



(a) Accuracy over Epoch

(b) Loss over Epoch

Figure 18. ResNet50 learning curves, on interpolated Falcon data in the all walks scenario with blue training and red validation.



(a)

(b)

Figure 19. Confusion Matrix of BasicNet trained on all Falcon data and tested on test file of all Falcon data. (a) Interpolated Falcon data (b) Non-interpolated Falcon Data

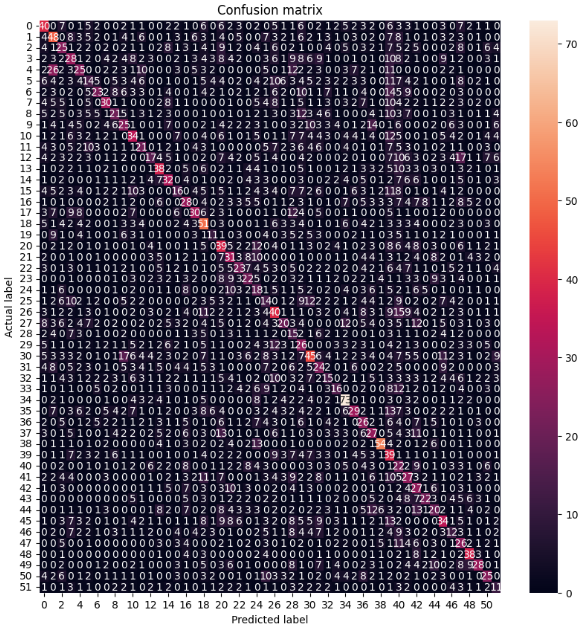


Figure 20. Confusion Matrix of BasicNet trained on all interpolated Velodyne data and tested on test data of all Velodyne data