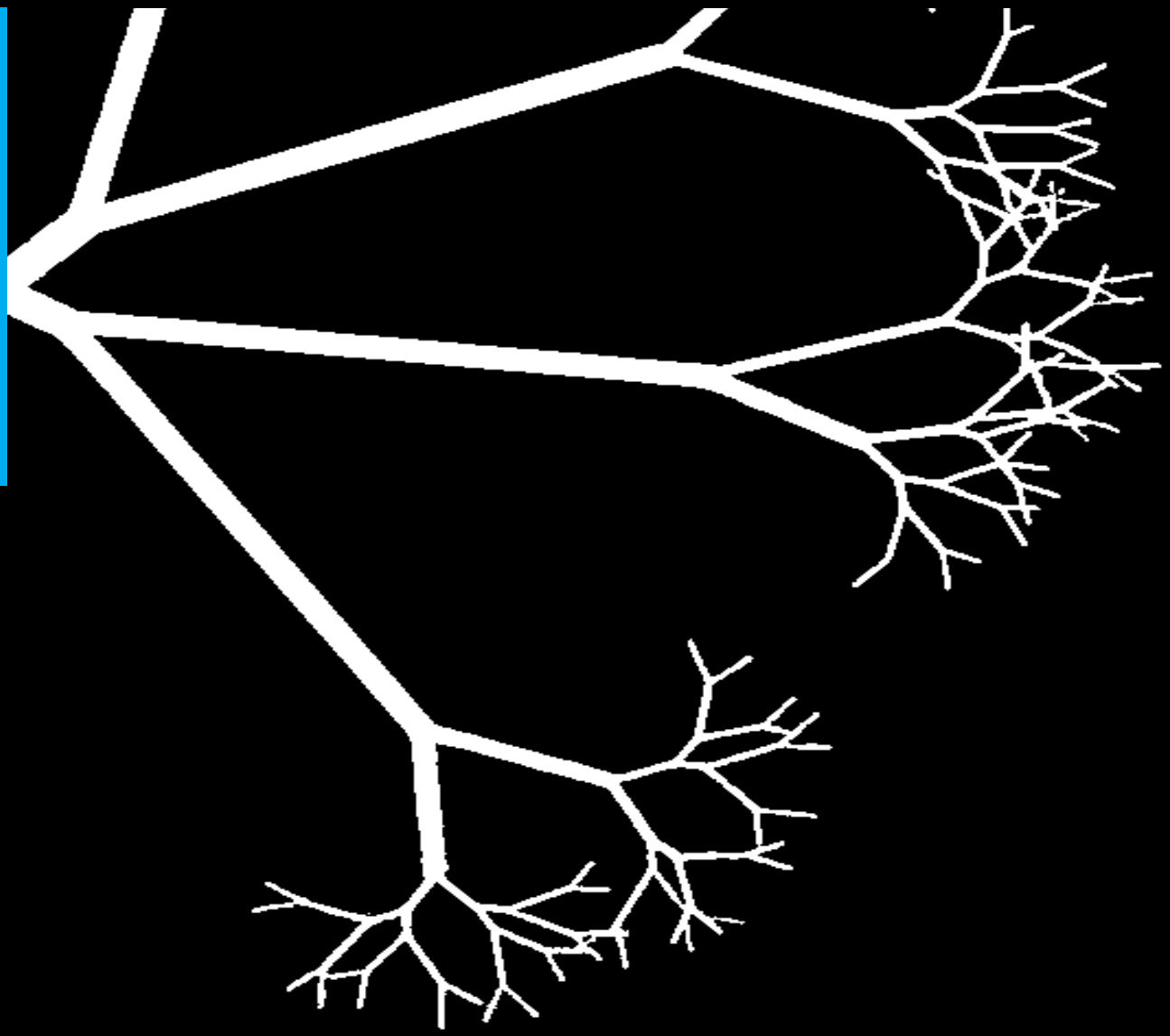
Reducing Ultrasound Localization Microscopy acquisition time for rat renal arterial tree using deeplearning model DBlink

Oscar He

MSc Thesis



TU Delft Delft University of Technology

Delft Center for Systems and Control

Reducing Ultrasound Localization Microscopy acquisition time for rat renal arterial tree using deeplearning model DBlink

by

Oscar He

For the degree of Master of Science in Systems and Control at Delft University of Technology

October 2, 2025

Supervisor(s): dr.ir. C.S. Smith
ir. K. Uğurlu

Faculty of Mechanical Engineering · Delft University of Technology

Abstract

Ultrasound imaging is a non-invasive imaging method, which uses ultrasound waves to produce images of the internal organs in the body. Since sound waves can interfere with each other, the ultrasound images are diffraction limited. Ultrasound localization microscopy (ULM) is a processing technique, which is able to bypass this diffraction limit by localizing individual spatially isolated contrast agent microbubble (MB)s in the low resolution ultrasound frames. These MBs acts as a point source and appears as a blurry point in the ultrasound frame also known as Point spread function (PSF) whose centroids can be localized with a precision beyond the diffraction limit. By localizing these MBs and tracking their paths over thousands of consecutive ultrasound frames and accumulating their tracks, a super resolution image of the vasculature can be reconstructed. While these super resolution images significant benefits to biomedical applications, they require long acquisition times.

This thesis investigates whether the deep learning model DBlink, a bidirectional convolutional long short-term memory (LSTM) with a Convolutional Neural Networks (CNN) head can reduce the long acquisition time of ULM. An in silico rat renal arterial tree was simulated to provide the data required for training and evaluating the deep learning model. Two different input type were explored for the DBlink model: Localization maps (summed frames of super resolved localizations) and velocity tracks (maps containing super resolved velocity tracks) of the MB. The effect of different receptive field (RFd) sizes were also examined.

The performance of the DBlink model was compared to the conventional ULM method and showed a reduced acquisition time of 8.7 seconds for large radii vessels in silico. However, the reduction in acquisition time diminishes for small radii vessel, where the passage of MB is still limited by low blood flow rate. Although DBlink reduces acquisition time, it introduces hallucinations in the reconstruction of vessels, especially in regions containing dense small vessels.

Overall, this research highlights the use of the deep learning model DBlink in ULM and the use of different input type to reduce the acquisition time in ULM. However, further research is needed in order to apply this deep learning model in vivo.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Ultrasound imaging is a popular and non-invasive imaging technique, which uses ultrasound waves to produce real time images of internal organs. These high frequency ultrasound waves are produced by the piezoelectric elements in a transducer device. When these ultrasound waves interact with tissues of different properties, these ultrasound waves are scattered back, resulting in a backscattered echo that can be captured by the transducer. By using a signal processing technique called beamforming, the captured signals of the backscattered echo can be converted to a ultrasound Brightness mode image or B-mode image.

Since sound waves can interfere with each other, there is a fundamental limit to the spatial resolution, which is the minimum distance which two objects can still be distinguished from each other. This spatial resolution limit is called a diffraction limit and is approximately half of the wavelength of the used wavelength [14]. The spatial resolution of a B-mode image can be improved by lowering the used ultrasound wavelength for example by using a higher utlrasound frequency, but this results into the ultrasound wave being attenuated faster when the ultrasound wave travels deeper into the body. This results in a trade-off between spatial resolutions versus penetration depth for a B-mode image.

In order to bypass the aforementioned trade-off, Ultrasound localization microscopy (ULM) technique has been developed which can be used to visualize the vasculature with a spatial resolution beyond the diffraction limit. ULM uses contrast-enhancing agent of air microbubble (MB) of a few micrometers in size which are injected intravenously. These MBs can be localized with a precision higher than the diffraction limit when these MBs are spatially isolated from each other. By localizing these spatially isolated MB over thousands of frames and tracking their paths over consecutive frames, a super-resolution image of the vasculature can be obtained by accumulating these MB tracks. Oncology and neurology can benefit strongly from these super-resolution images of the vasculature.
Although ULM can visualize the vasculature with a spatial resolution beyond the diffraction limit, it still needs to localize isolated MB over thousand of B-mode frames which can take a long time to be able to reconstruct one super-resolution image. This results in a trade-off between spatial resolution versus acquisition time or the temporal resolution.

Deep learning offers a variety of solutions in ULM to increase the temporal resolution. Most deep learning methods used in ULM, has been trained to be able localize MB positions in high MB concentration conditions.
In van Sloun *et al.* [15], they proposed a Convolutional Neural Networks (CNN) with an encoder-decoder structure named U-net, which is able to learn overlapping MB interference patterns and localize their MB positions. Their method used as input a single frame of 2-D low spatial resolution Radio Frequency (RF) data and outputs a high spatial resolution of the localization that is upsampled by a factor of 8 compared to the input. Following van Sloun [15], Milecki *et al.* [16] also used CNN with an encoder-decoder structure named V-net localize MB in high MB concentration conditions. Their deep learning model uses 3-D data, which are the sequences of B-mode ultrasound image frames or cineloops instead of a single frame used in van Sloun. The 3-D spatiotemporal data contained more information, such as the hemodynamics from the temporal data and also the vasculature shapes were taken into account in the simulation of the ultrasound data. From the spatiotemporal input data, the network improved the localization in high MB concentration conditions.
Although van Sloun *et al.* [15] and Milecki *et al.* [16] were able to localize MB in high concentration conditions, multiple tracks of MBs were still needed to reconstruct the blood vessels.

In Chen *et al.* [17], their deep learning model was able to output the super resolution reconstruction of the vasculature structure and velocimetry directly using less MB localizations than needed in conventional ULM, this is possible due to the model having learned about the structure shapes of the organ it was trained on. The model has a CNN encoder-decoder structure U-net, but with an long short-term memory (LSTM) cell in the bottleneck of the structure to be able to learn spatiotemporal data from the B-mode cineloops. The model was trained to output a super-resolution velocimetry of the vascular structure using 16 frames of B-mode images, which corresponds to a temporal resolution to a temporal resolution of $0.016s$. Although this method has a high temporal resolution, the model hallucinated fake vessels which did not exist where there were not enough MBs present for example at small vessels which compromises the spatial resolution.

Recently, in Saguy *et al.*[8] a model known as DBlink used for single-molecule localization microscopy (SMLM), which has a bi-directional convolutional LSTM connected in series with a CNN. The model used as input super-resolved summed MB localizations known as localization maps which was used as input to predict directly the super resolution images of filaments and mitochrondia-like structures which resulted in low acquisition time and low hallucination rate.

1-1 Research objective

With the need to reduce the acquisition times of ULM without sacrificing spatial resolution, deep learning offers a promising way to increase the temporal resolution. Van Sloun [15] and Milecki *et al.* [16] offers a deep learning method to be able to localize MBs in high MB concentration conditions. Chen *et al.* [17] was able to reconstruct the velocimetry of the vasculature structure using low acquisition time, but hallucinated vessel in locations where there were not enough MBs present for example small radius vessels, which compromises the spatial resolution.

In this research the goal is to continue the steps of Chen *et al.* in reconstructing the full vasculature structure super-resolution image directly with reduced acquisition time and low hallucination by leveraging DBlink model on ULM data and evaluate its performance of reconstruction of a simulated synthetic rat renal artery tree against the conventional non deep learning ULM.

1-2 Research questions

The research questions along with its sub-questions for this master thesis are as following :

- 1. How can the DBlink model be trained for ULM?
  - How to simulate ULM data for the model?
  - What loss functions and metrics for training and evaluating the model ?
  - Does size of the simulated structure affect the performance of the model?
  - How does different ULM inputs affect the model?
- 2. Does the trained DBlink model reduce acquisition time when compared to the conventional ULM method?

To answer research question one, ULM data will be obtained by simulating synthetic rat renal artery tree structure and trained on two different DBlink models with each having different receptive field (RFd) size to assess effect of the large renal tree structure and finally assessing the effect of using different input of localization maps versus velocity maps.
To answer research question two, the best performing model obtained from research question one will be benchmarked against conventional ULM method.

In Chapter 2 the background for ULM and deep learning are provided. The methodology for simulating dataset and training models are explained in Chapter 3 and the results are shown in Chapter 4. The results are discussed in Chapter 5. Finally in Chapter 6 conclusions are drawn along with limitations and future research recommendations.

Chapter 2

Background

In this chapter the background knowledge for conventional understanding ultrasound imaging is provided in Section 2-1. To bypass the diffraction limit in ultrasound imaging and how a super resolution image can be obtained using the Ultrasound localization microscopy (ULM) processing are shown in Section 2-2. Additionally in Section 2-3, the deep learning background knowledge to understand the DBlink model used in this thesis.

2-1 Ultrasound Brightness mode images

2-1-1 Ultrasound waves

In order to understand how ultrasound images are obtained, it is first necessary to understand what the ultrasound waves are and how it interacts with the surrounding tissues in the body. Ultrasound waves are sound waves with frequencies higher than the upper limit of human hearing. The frequency range which humans can hear is between 20 Hz and 20 kHz, the frequency range used in medical imaging is between 1 MHz and 20 MHz [18]. The ultrasound waves used for medical image scans are longitudinal waves, see Figure 2-1. When these wave travel through an elastic medium, the particles in a region of the medium will experience an alternating sequence of compression and decompression which is also known as rarefaction, with compression corresponding to positive pressure and negative pressure for rarefaction. The distance between two positive peak pressures is given by [1]:

$$\lambda = \frac{c}{f}, \tag{2-1}$$

where λ is the wavelength, c the speed of sound in the medium and f the frequency of the ultrasound wave. Given a constant speed of sound, the wavelength is then determined by the ultrasound wave frequency. The speed of sound in soft tissue is 1540 m/s and in fatty tissue 1450 m/s [18].

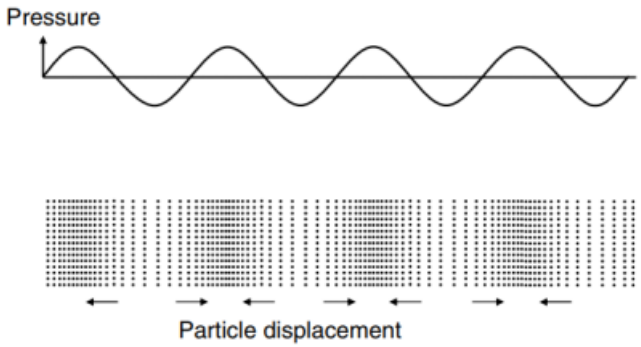


Figure 2-1: A longitudinal wave with particles moving horizontally. Source: [1].

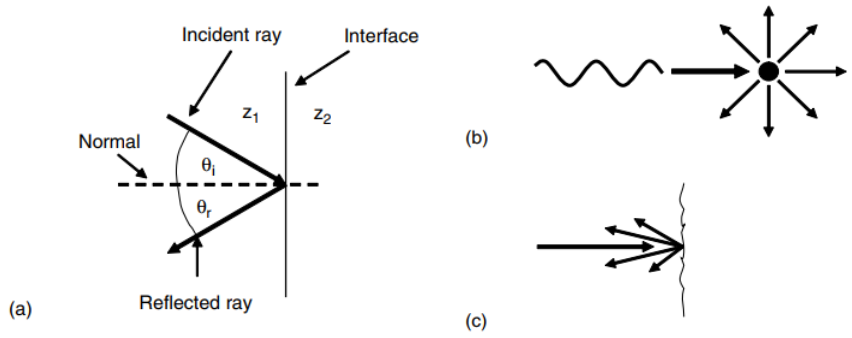


Figure 2-2: Ultrasound waves reflected by large interface and scattered by small targets. (a) At a large smooth interface the reflected wave angle is the same as incident angle. (b) Rayleigh targets scattering the wave in all angles. (c) Diffuse reflection by a rough surface. Source: [1].

The pressure which the medium experiences when ultrasound wave is propagated through it is given by [19]:

$$P = \rho c u_{\mathrm{p}} = Z u_{\mathrm{p}}, \tag{2-2}$$

where ρ is the medium density and u_p is the particle velocity and Z the acoustic impedance.

Ultrasound images are generated through the different interactions of sound with the tissues. When sound is propagated through a medium and encounters another medium with different acoustic impedance the wave will be reflected, refracted or scattered depending on the properties of the encountered medium.

In reality some of the surfaces in the organs may be slightly rough on the scale of the wavelength such as the liver [18], reflection here does not follow the law of reflection and instead the wave will be scattered in different range of angles which is known as diffuse reflection, as inFigure 2-2(c). For pointlike targets where the size of the interface is much smaller than the sound wavelength λ, the wave will be scattered uniformly in all directions surrounding the target with a shape of spherical wave as in Figure 2-2(b), this is also known as Rayleigh scattering.

Attenuation

When ultrasound waves travels through the tissue, the amplitude of the pressure in the wave will be gradually attenuated with distance travelled. This is caused by absorption, reflection and scattering when the wave travels through the tissue [1]. Absorption is when the energy in the wave is converted into heat in the medium. The rate at which the amplitude of the wave decays is exponential in nature. The pressure at a distance from the source pressure can be approximated by given by [19]:

$$P(z, f) = P_0 e^{-af^b z}, \tag{2-3}$$

where P_0 is the pressure at the transducer source, z the distance traveled from the source and a, b are empirical constants depending on the type of tissue. Eq. (2-3) shows that waves with higher frequency the amplitude will decay faster than lower frequency waves.

2-1-2 Ultrasound Brightness mode imaging

Brightness mode image or B-mode image is an ultrasound image where the brightness of the pixel corresponds to the amplitude of the received echo. In order to understand how B-mode images are obtained using ultrasound, a drawing is shown on Figure 2-3. First in Figure 2-3(a), a transducer with a linear array produces a plane wave by sending a short pulse at the same time for all elements (typically 128 transducer elements [2]) in the array.

When the wave travels through the medium, the medium will scatter echos back to the transducer due to impedance mismatch in the medium. Assuming that the medium is composed of pointlike rayleigh backscatters, the echo will be a spherical wave. When the backscattered echo come into contact with an element of the transducer, the pressure from the wave on the element will then record the Radio Frequency (RF) signal as seen on Figure 2-3(b). With each element having a recorded time series of RF signal.

To create the B-mode image, the RF signals has to be beamformed for each pixel in the B-mode image corresponding to a location in the scanned medium as shown on Figure 2-3(c). Beamforming is done by delaying the RF signals coming from each transducer element and then summing the signals, since a spherical wave arrives at each transducer elements in different time intervals. Beamforming has an effect of focusing on the incoming spherical wave from a specific location in the medium.

2-1-3 Spatial resolution

Since sound waves can interfere with each other, there is a fundamental limit to how much a transducer can focus on a point in the medium. This limit is called the diffraction limit and is not unique to sound but also other wave-based imaging processing techniques in optics [3]. Due to the diffraction limit there is a theoretical limit on the spatial resolution of the ultrasound image. The ultrasound image focused on a point scatterer will then appear as a blurry point, this blurry point is also know as a Point spread function (PSF).

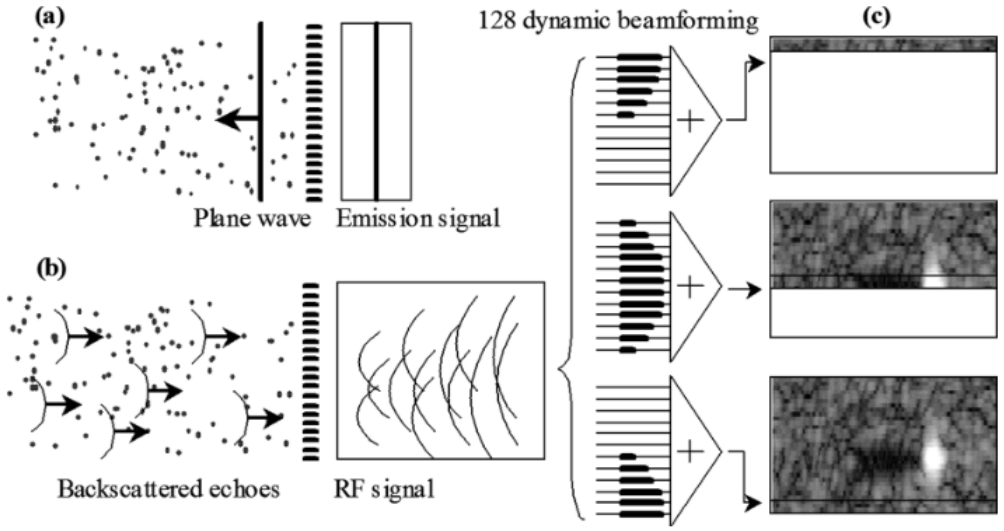


Figure 2-3: Overview on the process of obtaining B-mode images. (a) A single plane wave pulse is transmitted by a linear transducer array to insonify the medium. (b) The scatterers in the medium backscatters spherical echos back to the transducer. (c) The RF signal is then beamformed by delaying RF signal from each individual transducer element and summed to create a B-mode image. Source: [2].

One way to express the spatial resolution of an image is the minimum distance which two separate PSF can still be distinguished from one another. A theoretical limit for the spatial resolution is given by the Rayleigh criterion [20]:

$$R_L = 1.22\lambda f_{\#} \tag{2-4}$$

where R_L is the minimum distance between two PSF that can still be resolved from one another, λ the wavelength of the ultrasound wave and $f_{\#}$ the ratio between depth and aperture receive size of the transducer array.

Another metric for resolution limit is the full-width at half maximum (FWHM) which is the width between the points on one side to the other side of the PSF where the intensity is at half of its maximum value, which is given by[14]:

$$\text{FWHM} = 1.4\lambda f_{\#} \tag{2-5}$$

From Eq. (2-4) and Eq. (2-5), it might seem straightforward to increase the spatial resolution by making the ultrasound wavelength λ smaller. But from Eq.(2-1) to obtain a lower wavelength, a higher ultrasound wave frequency is needed. However from Eq.(2-3) the wave will be attenuated faster with a higher frequency, so a trade-off has to be made between resolution and depth of the ultrasound scan.

2-1-4 Temporal resolution

The rate at which a B-mode image is acquired depends on the travel time of pulse echo sequence. The standard method to insonify the medium is line by line, in which a narrow

beam that is beamformed is transmitted in a line in front of each transducer element. This process is time-consuming, since a B-mode image is made up typically of 100 or more of B-mode lines with frame rates up 30-60 Hz [1].

By using only a plane wave to insonify the whole medium as seen on Figure 2-3, B-mode images can be acquired with frame rates in the kHz range. With the method of coherent plane-wave compounding [2], a series of B-mode images were obtained by scanning the medium with a range of angled plane-waves. The B-mode images were then compounded together to obtain a single B-mode image with a higher spatial resolution than the individual images.

2-2 Ultrasound Localization Microscopy

2-2-1 Introduction to ULM technique

ULM is a super-resolution ultrasound technique that can visualize vascular structure and flow beyond the diffraction limit that ultrasound B-mode suffers from. ULM uses a contrast-enhancing agent microbubble (MB) of a few micrometers in size [14] and is injected intravenously. These MBs are used to make an image of blood vessels in ultrasound imaging due to their high impedance mismatch of blood and air and also acting as a resonator in the resonance frequency range of 1-15 MHz [3].

The PSF of the MB can be localized with a precision far higher than the diffraction-limited resolution [3], with the prerequisite that the MBs is isolated from other such that their PSF they do not interfere with each other, since this degrades the precision of their localizations. ULM exploits the localization of these isolated MBs by collecting their localizations over thousands of frames to create a super-resolution image of a vascular structure. This method of localizing spatially isolated PSF of MB and accumulating localization to create a super-resolution image has been inspired by single-molecule localization microscopy (SMLM). In SMLM fluorescent molecules are used on a specimen instead of MB, which only a sparse random subset of molecules emits light in a single frame when captured.

2-2-2 ULM processing pipeline

The general processing pipeline for ULM technique is shown on Figure 2-4, which shows the steps from the acquisition of ultrasound images to the visualization of the super-resolution image.

The first step is the acquisition of ultrasound B-mode frames over time of the contrast-enhanced vascular region, where the injected flows MBs flows through. From the ultrasound frames, detection and isolation of the MB is performed. Detection is a crucial part of the pipeline, which is separating the MBs from the surrounding tissues and creating candidate bubble regions for localization.

Following the detection, is the isolation of MBs, where a filtering step identifies isolated MBs in each ultrasound frame and rejects interfering PSF of MBs that are closer to each other than the diffraction limit. This step is necessary since interfering PSFs degrades the localization accuracy.

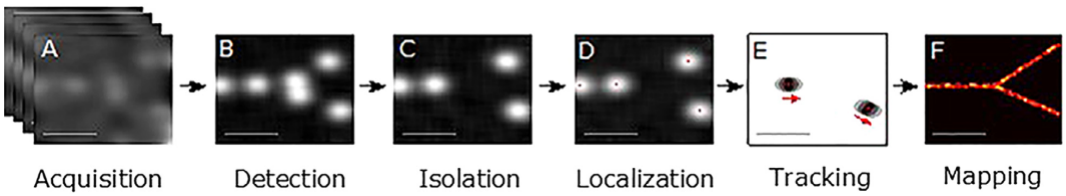


Figure 2-4: Steps in super-resolution ultrasound processing. (a) The acquisition of ultrasound frames from contrast-enhanced vascular region with MB. (b) Detection of MB signals. (c) Isolating individual MB. (d) Localization of MB with a precision greater than the diffraction-limit resolution. (e) Tracking of MBs over consecutive frames to obtain velocity profiles. (f) Mapping of the accumulated localizations to construct a super-resolution image. Source: [3].

Next step is the localization of the MBs, where the location of the isolated MBs is estimated with a precision far higher than diffraction resolution limit. Some methods for this step is shown in Section 2-2-3.

Tracking of the MBs can be performed after the localization of MBs. Because the MBs are flowing through the vascular system over time, the displacement of a MB can be tracked from two consecutive images. From the displacement of the MB, the path and velocity vector can be determined. This step is explained in Section 2-2-4

Finally, the last step is the visualization of the accumulated localization and tracks of the MBs over a series of frames producing a super-resolution image.

2-2-3 Localization

For the localization of MBs, their locations can be obtained from both the RF or beamformed signals [3]. Both ways use the travel time of the spherical echo from MB to the transducer element to determine the MB localization.

Localization with RF signal

For localization using RF data, the echo of a MB will appear as a parabola ,with the center of the summit summit corresponding to the MB location [21]. To find the MB location, a parabola is fitted to the spherical echo from the MB. Then the center of the summit of the fitted parabola is used to extract the axial and lateral positions of the MB with the calculation of the echo travel time.

Localization with beamformed signal

For localization of MB on beamformed data, the MB will appear as a blur or PSF with the location of the MB corresponding to the center of the PSF [3]. One way to find the center is by calculating the intensity-weighted center of mass [22]:

$$[C_x, C_y] = \frac{\sum_i I(x_i, y_i)[x_i, y_i]}{\sum_i I(x_i, y_i)}, \tag{2-6}$$

where C_x and C_y is the x and y locations respectively of the center. $I(x_i, y_i)$ is the pixel intensity value at location (x_i, y_i).

Another way is to fit a Gaussian function to a PSF [3], since they are similar to each other in the shape. This can be done by fitting 2-D Gaussian function with fixed amplitude and standard deviation to a PSF using a least squares method[23]. The amplitude and standard deviation is empirically determined by the expected response of a MB and the center of the fitted Gaussian will then correspond to the MB location. The fitting can also be done by using cross-correlation [24] or deconvolution [25] of a fixed variable 2-D Gaussian function with the PSF to find the MB location directly.

Finally a simple method, is the peak detection, where the peak intensity value pixel of the PSF is the location of the MB [23]. A way to enhance this simple, is to use an interpolation-based algorithm to upscale the current pixels and finally detect where the peak intensity lies on those subpixels. Some of the most used interpolation algorithm used are: Lanczos, Spline and Cubic interpolation [4].

2-2-4 Tracking

The tracking of MB involves in finding the track of individual MBs, which allows the velocity and direction of the flow to be mapped in the super-resolution image. By tracking MBs, it also has the added benefit of rejecting false MB localizations by allowing only coherent paths and rejecting short paths [3].

In early tracking methods used to track MBs, simple tracking algorithms were used. In Christensen *et al.* [26] the PSF signal of indivual MBs were cross-correlated with PSFs of MBs in the upcoming ultrasound frame within a certain distance window. In Errico *et al.* [25] the closest-neighbour detector was used where the closest MB in the next frame was considered to be the same MB in the previous frame. More advanced algorithms are currently used to track MBs such as using Markov Chain Monte Caro and Kalman filter were used [3].

2-2-5 ULM resolution

Spatial resolution

By localizing the isolated MB position with a precision higher than the diffraction limit, a super-resolution image with a high spatial resolution can be made. The spatial resolution limit of the image is dependent on the precision which the MB is localized. This precision is in turn dependent on the precision of the estimated time delay of the RF signal for the individual transducer elements during beamforming. In Desailly *et al.*[27] a model was developed which can predict the estimated lateral and axial theoretical resolution limit based on the time delay precision:

$$\sigma_{\hat{x_0}} \approx 2\sqrt{3}\frac{c\sigma_\tau z_0}{\sqrt{n}L_x}, \tag{2-7}$$

$$\sigma_{\hat{z_0}} \approx \frac{c\sigma_\tau}{2\sqrt{n}}, \tag{2-8}$$

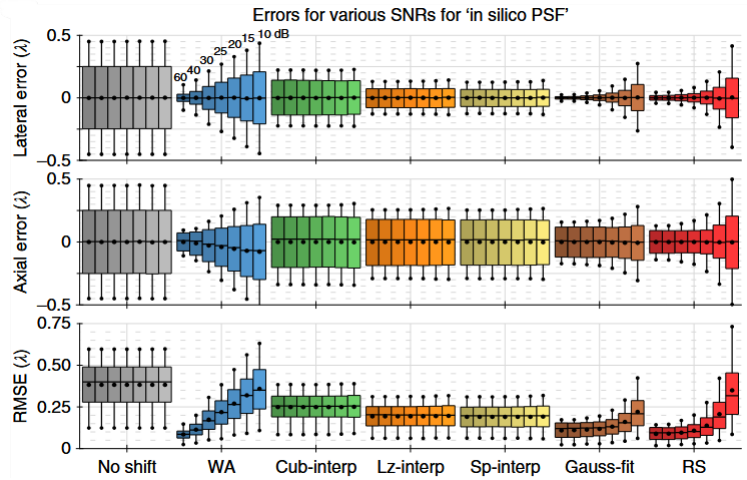


Figure 2-5: Lateral, axial error and RMSE of different algorithms in terms of wavelength λ for different SNR scenarios. Where black dots represents the average, upper and lower boxes the upper and lower quartiles and finally the whiskers the 5% and 95% quantiles. Algorithm names from left to right, No Shift, Weighted Average, Cubic interpolation, Lanczos interpolation, Spline interpolation, Gaussian-fit and Radial symmetry. Source [4].

where $\sigma_{\hat{x_0}}$ and $\sigma_{\hat{z_0}}$ is the lateral and axial resolution respectively. c is the speed of sound, L_x is the lateral length of the aperture, n is the number of transducer elements in the aperture, z_0 is the axial length from transducer element to the scatterer and lastly σ_τ is the standard deviation of the time delay. σ_τ can be measured directly or calculated using the equation provided in [27].

In practice, the spatial resolution is also influenced by more factors, such as the Signal to Noise ratio (SNR) of the PSF and the localization-algorithm that is used to find the centroid of the MB. In Heiles *et al.*[4] a benchmark was done on the performance of several commonly used localization-algorithms used on beamformed ultrasound frames. In the study an in silico data was generated where a MB was beamformed in a $\lambda \times \lambda$ pixel size grid where the MB moved by $\frac{\lambda}{21}$ steps each time with different SNRs ratio. From the benchmark result as seen on Figure 2-5, the lateral and axial resolution pixel size can be estimated depending on the localization-algorithms and SNR.

Temporal resolution

The temporal resolution of ULM is limited by several factors. Firstly, to localize MB, their PSF need to be spatially isolated from each other to be able to be localized with a precision beyond the diffraction limit. This means that there is a limit to the concentration of MBs in the blood vessels, meaning there is a limit on how many MBs can be localized from a single ultrasound frame.

The second limiting factor is that multiple tracks of MBs are required to construct a vessel, since a large vessel cannot be represented by a single MB track.
Hingot *et al.*[28] proposed that the minimum amount of tracks required to reconstruct a vessel as the the width of the vessel divided by the pixel size of the super-resolution image:

$$N = \frac{d}{l_{pix}} \tag{2-9}$$

where N is the number of pixels necessary to reconstruct a vessel of diameter d with pixel size of l_{pix}.

Since the blood flow in vessels follows the Poisseuille law [28], the flow can be described as:

$$Q(d) = \frac{\pi d^4}{128\eta}\frac{\Delta P}{L} \tag{2-10}$$

where η is the blood viscosity, ΔP the continuous pressure drop of the blood vessel and L the length of the vessel.

The number of MB that passes through a blood vessel is given by:

$$N = Q(d) \cdot C_{bubbles} \cdot T_{acq} \tag{2-11}$$

where N is the number of MB passing through a vessel, $C_{bubbles}$ the MB concentration in bloodvessels and T_{acq} the acquisition time.

By substituting Eq. (2-9) along with Eq. (2-10) into Eq. (2-11), the minimum theoretical acquisition time T_{acq} to construct a vessel with a diameter d is then given by:

$$T_{acq} = \frac{1}{l_{pix} C_{bubbles} \frac{\pi d^3 \Delta P}{128\eta L}} \tag{2-12}$$

With Eq. (2-12) the proportionality between T_{acq} and d is then given by:

$$T_{acq} \propto d^{-3} \tag{2-13}$$

which means that small-diameter vessels need a lot more acquisition time to reconstruct compared to larger-diameter vessels. This also shows that there is a trade-off between the temporal resolution and spatial resolution for a super-resolved image, since more time is required to reconstruct a high spatial resolution image of the small blood vessels like the capillaries. In Hingot *et al.*, it was shown that a large diameter vessel larger than 100 micrometers can be fully reconstructed in 10 seconds while small capillaries will take tens of minutes to reconstruct.

2-2-6 Biomedical applications of ULM

Since ULM is a non-invasive method that can make a super-resolution image of the vasculature, it can be used in two main fields that are closely related to the study of the vasculature. The two main fields that can benefit strongly is oncology and neurology.

In oncology, the early detection of malignant lesions caused by cancer can increase the chance of successfull treatment [3]. One of the early signs of cancer is the formation of new malignant blood vessels, which is known as malignant angionesis. The detection of the malignant angionesis with the traditional B-mode images is difficult with its limited spatial resolution due to the fundamental diffraction limit which can be overcome through the use of ULM.

With the ability to resolve deep microvasculature, ULM can also be applied in neurology, where it can be used for both diagnostics and therapeutics for the brain. The pathological process of small vessels in the brain is a contributor to disorders such as stroke and Alzheimer disease.

The application of ULM for these fields are still in the pre-clinical stages and has been mainly applied only on animals [29]. There are several major challenges that limits use of ULM on humans. One of the major challenges is the long scan times to reconstruct a super-resolution image which can take several minutes to reconstruct microvasculatures. And due to the long scan times, it is also inevitable that the subject moves during the scanning which is caused by the scanner probe movement, respiration, cardiac cycle and other unavoidable sources of motion which causes a combination of translation, shearing and nonrigid deformations of the tissue [30]. These movements introduce motion artifacts which limits the spatial resolution rather than being limited by the precision of the MB localizations.

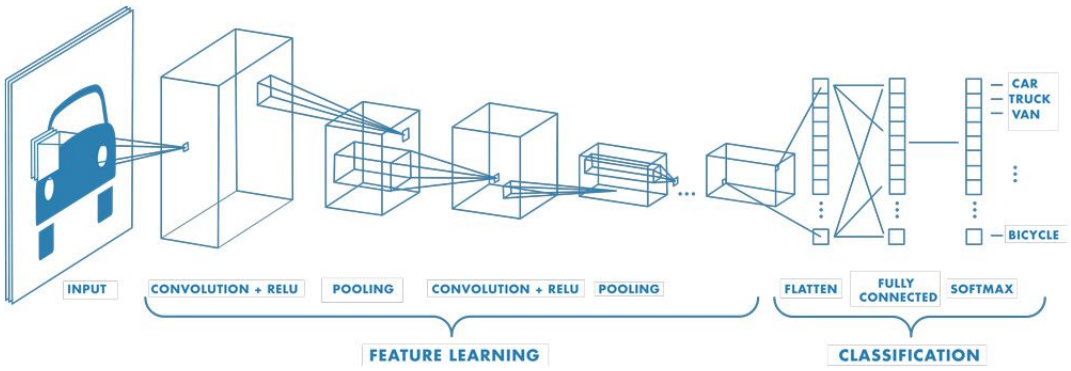


Figure 2-6: A CNN example used for classification of an image that consists of convolutional, pooling, and fully-connected layers. The features from the input image are extracted by the convolutional layer and are then downsampled by the pooling layers. The convolutional and pooling layers can be sequentially repeated to extract more abstract features. The final features from are then flattened to be fed into the fully-connected layers with a softmax function in the output layer for image classification task. Source: [5].

2-3 Convolutional Neural Network

Convolutional Neural Networks (CNN) are neural networks that are specialized to process datas that has a grid-like topology such as images and are more computationally efficient compared to multilayer perceptrons (MLPs), since CNN uses less number of weight parameters with the use of convolution. The CNN for an image classification typically consists of convolutional, pooling ,and fully-connected layers [31]. For semantic segmentation where all pixels are classified rather than the image as whole, the fully-connected layer is left out.

In Figure 2-6 an example structure of a CNN that is used for image classification task is shown. The input image is first passed through a convolutional layer which extracts features from the input and outputs a feature map. The feature map is then passed through an activation function, where nonlinear features can be extracted and after the activation function the output is passed through pooling layer where the feature map is downsampled, this helps in achieving shift-invariances and also reducing the computations required in the CNN.

The convolutional layer can be applied multiple times in sequence, where in the the first sequence the low-level features such as edges and curves are extracted. While in the deeper layers more abstract features are extracted from the low-level features. At the end of the sequences, the output is flattened into a 1-D data and fed through one or several fully-connected layers, where the outputs of the neurons in these layers are connected to all the neurons from the previous layer forming a fully-connected layer. In the output layer of the CNN, an activation function is applied such as the sigmoid function which is used for binary classification task.

2-3-1 Convolutional layer

In the convolutional layer, the main objective is to learn the feature representations from its inputs. There can be one or several kernels or filters used for convolution and computing the

feature maps for the output as shown on Figure 2-7. For a 2-D input data, the feature map value at (i, j) in the k'th kernel of the l'th layer is given by [31]:

$$z^l_{i,j,k} = {\mathbf{w}^l_k}^T \mathbf{x}^l_{i,j} + b^l_k, \tag{2-14}$$

where $\mathbf{w}^l_k$ and b^l_k are the weight vector and bias respectively of the k'th kernel in the l'th layer. And $\mathbf{x}^l_{i,j}$ is the input patch centered at location (i, j) in the l'th layer wich has the same size as the kernel.
After the convolution, non-linear activation function can be applied on the feature maps to extract non-linear features from the input. The activation value $a^l_{i,j,k}$ is then given by:

$$a^l_{i,j,k} = \sigma(z^l_{i,j,k}), \tag{2-15}$$

where σ is the activation function, typical activation functions used are the Sigmoid, Tanh and ReLU function [31].

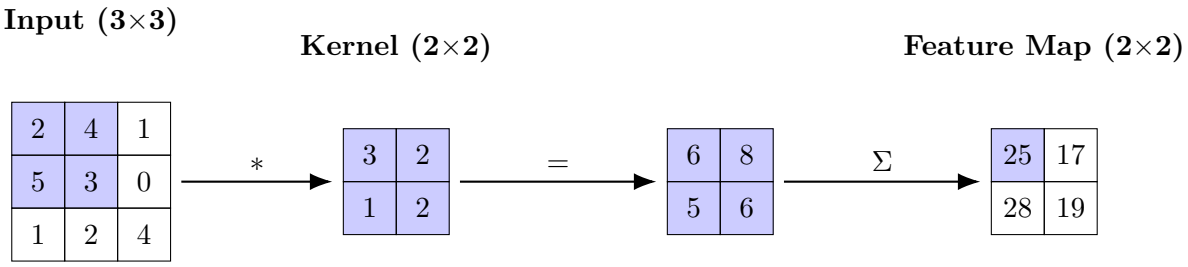


Figure 2-7: 2-D Convolution example of a 2x2 kernel with stride of 1 on a 3x3 input to obtain a 2x2 feature map.

The spatial input area which the feature map pixel is calculated from or "sees" is the receptive field (RFd), in the case of Figure 2-7 the RFd size is 2. The RFd size can be increased by repeating the convolutional layer sequence multiple times. The RFd size can be further increased by using dilated convolution, which introduces a gap during the convolution as seen on Figure 2-8.

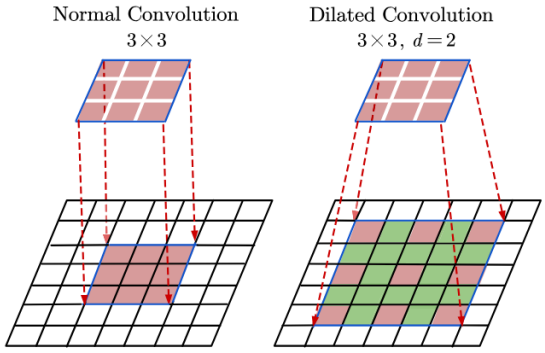


Figure 2-8: A dilated convolution comparison with the regular convolution using a 3x3 kernel. In the Dilated convolution with a dilation factor of $d = 2$, the input area which the convolution is operated on has gaps shown in green. Normal convolution has a dilation factor of $d = 1$. Source: [6].

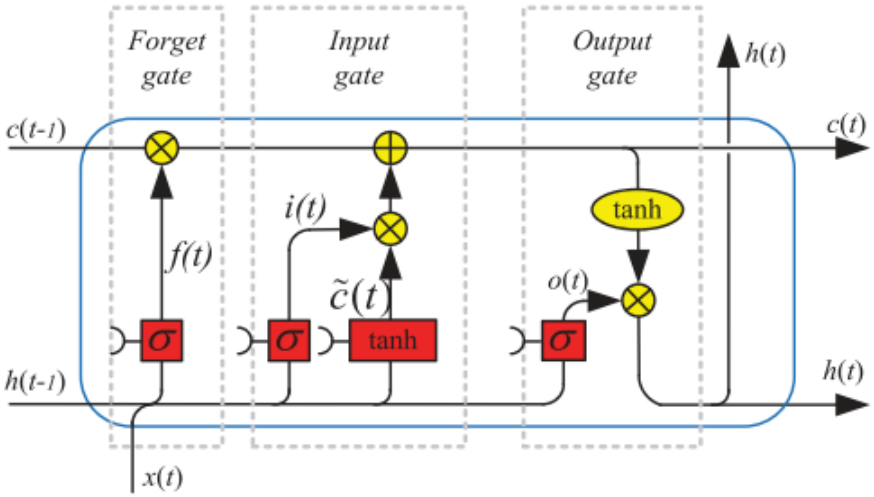


Figure 2-9: Schematic illustration of the standard LSTM cell that has a forget, input, output gates with a new variable cell state c compared to the stand recurrent cell. Source: [7].

2-4 Recurrent Neural Network

Recurrent Neural Networks (RNN) are a type of neural networks that are specialized in processing sequential data. In a RNN network, the output at the current time step are calculated as a function of the current input and a recurring information from the previous time step.

2-4-1 LSTM

The long short-term memory (LSTM) cell is a popular RNN cell that uses gates to overcome the issue of long-term dependencies which causes the learning gradients to explode or vanish. The schematic illustration of the standard LSTM cell is shown on Figure 2-9. The mathematical expressions for the standard LSTM cell is given by [7]:

$$\begin{aligned} f_t &= \sigma\left(W_{fh}h_{t-1} + W_{fx}x_t + b_f\right), \\ i_t &= \sigma\left(W_{ih}h_{t-1} + W_{ix}x_t + b_i\right), \\ \tilde{c}_t &= \tanh\left(W_{\tilde{c}h}h_{t-1} + W_{\tilde{c}x}x_t + b_{\tilde{c}}\right), \\ c_t &= f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t, \\ o_t &= \sigma\left(W_{oh}h_{t-1} + W_{ox}x_t + b_o\right), \\ h_t &= o_t \cdot \tanh\left(c_t\right), \end{aligned} \tag{2-16}$$

where c_t is the cell state of LSTM which can be thought of as the long term memory and the hidden state h_t can be thought of as the short term memory. f_t, i_t and o_t is the gate values for the forget, input and output gates respectively. W_f, W_i, W_o are the weights and b_f, b_i, b_o are the bias for the aforementioned gates. Here the activation function σ is a sigmoid function because the values are between 0 and 1.

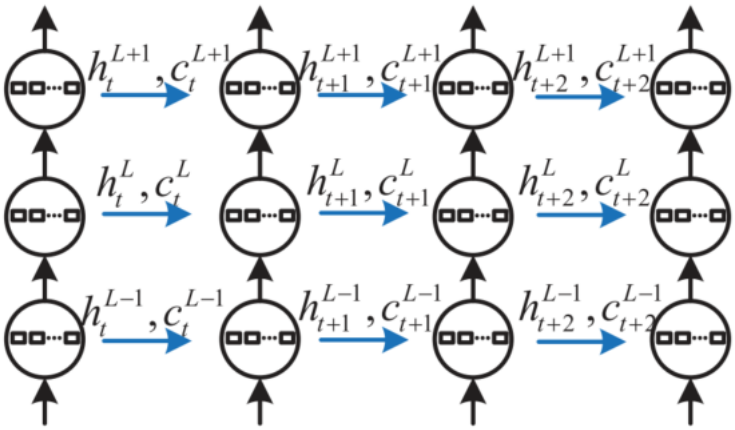


Figure 2-10: A schematic illustration of a stacked LSTM structure with 3 layers of depth. Source: [7].

The forget gate determines what information is thrown away from the cell state, with 0 everything thrown away and 1 all being retained. Similarly, the input gate decides on how much information is coming in from the input into the cell state and the output gate decides on how much information from the cell state is going to the output.

To add capacity in the deep learning model in order to learn tasks with more complexity, the LSTM cells can be stacked as shown on Figure 2-10. The new mathematical expression for a LSTM cell in the Lth layer is given by [7]:

$$\begin{aligned}
f_t^L &= \sigma\left(W_{fh}^L h_{t-1}^L + W_{fx}^L h_t^{L-1} + b_f^L\right), \\
i_t^L &= \sigma\left(W_{ih}^L h_{t-1}^L + W_{ix}^L h_t^{L-1} + b_i^L\right), \\
\tilde{c}_t^L &= \tanh\left(W_{\tilde{c}h}^L h_{t-1}^L + W_{\tilde{c}x}^L h_{t-1}^L + b_{\tilde{c}}^L\right), \\
c_t^L &= f_t^L \cdot c_{t-1}^L + i_t^L \cdot \tilde{c}_t^L, \\
o_t^L &= \sigma\left(W_{oh}^L h_{t-1}^L + W_{ox} h_{t-1}^L + b_o^L\right), \\
h_t^L &= o_t^L \cdot \tanh\left(c_t^L\right),
\end{aligned} \tag{2-17}$$

where the superscript L indicates the layer of the LSTM cell.

Convolutional LSTM

In order to be able to process sequential spatial data such as images in a video, convolution can be used in the LSTM cell which is called a convolutional LSTM. In Figure 2-11 a schematic illustration of a convolutional LSTM is shown. In a convolutional LSTM cell, a convolutional operator is used now along with kernel weights to calculate the future states of the cell, the mathematical expressions for a convolutional LSTM cell is then given [7]:

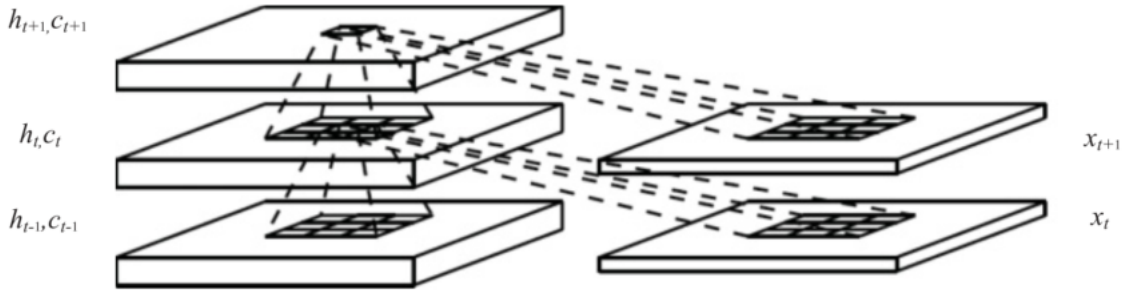


Figure 2-11: A schematic illustration of a convolutional LSTM cell. Source: [7].

$$
\begin{aligned}
f_t &= \sigma\left(W_{fh} * h_{t-1} + W_{fx} * x_t + b_f\right), \\
i_t &= \sigma\left(W_{ih} * h_{t-1} + W_{ih} * x_t + b_i\right), \\
\tilde{c}_t &= \tanh\left(W_{\tilde{c}h} * h_{t-1} + W_{\tilde{c}x} * x_t + b_{\tilde{c}}\right), \\
c_t &= f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t, \\
o_t &= \sigma\left(W_{oh} * h_{t-1} + W_{ox} * x_t + b_o\right), \\
h_t &= o(t) \cdot \tanh(c(t)),
\end{aligned}
\tag{2-18}
$$

where '$*$' is the convolutional operator and '$\cdot$' the Hadamard product.

Chapter 3

Methodology

In this chapter the methodology of the thesis is laid out. Section 3-1 introduces the DBlink deep learning model and the proposed models for Ultrasound localization microscopy (ULM). Section 3-2 describes the the pipeline for the data simulation of the renal arterial tree. Lastly, Section 3-3 explains the loss and performance metrics along with the strategy to optimize the hyperparameters of the proposed models.

3-1 Model setup

3-1-1 DBlink deep learning method

DBlink is a deep learning method which was used on single-molecule localization microscopy (SMLM) to reconstruct a super-resolution image with a low acquisition time and high spatial resolution using a deep learning model which will be known as DBlink. DBlink model uses Bi-directional long short-term memory (LSTM) cells connected in series with Convolutional Neural Networks (CNN) head to combine the future and past information coming from the Bi-directional LSTM cells as shown on Figure 3-1.
DBlink uses as input a sequence of super-resolved localizations maps, which consists of super-resolved localized frames which are then summed from N_{sum} as shown on Figure 3-2 and the model outputs a super-resolved image for each provided localization map.

This method is able to achieve high spatiotemporal resolution because of two main factors. Firstly, the structural information coming from the specific structure of a specimen it was trained on, which contained a distinct structure shape. Secondly the use of Bi-directional LSTM structure not only provides information coming from a single localization map, but also the short and long term information contained from the future and past localization maps.

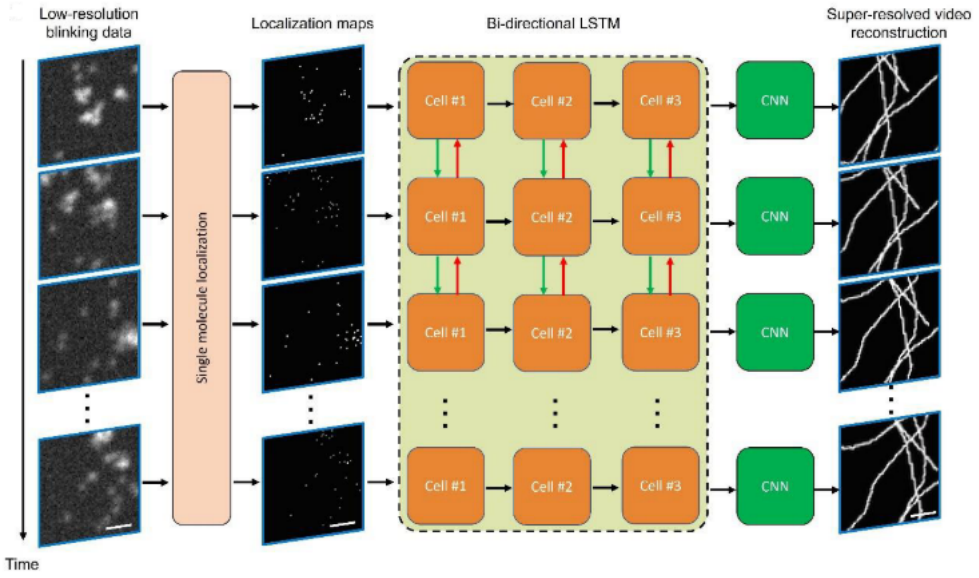


Figure 3-1: DBlink deep learning structure compromised of a Bi-directional LSTM connected in series with a CNN head. DBlink uses as inputs a sequence of super-resolved localization maps and outputs a super-resolved video reconstruction of a microtubule filament structure in SMLM. Source: [8]

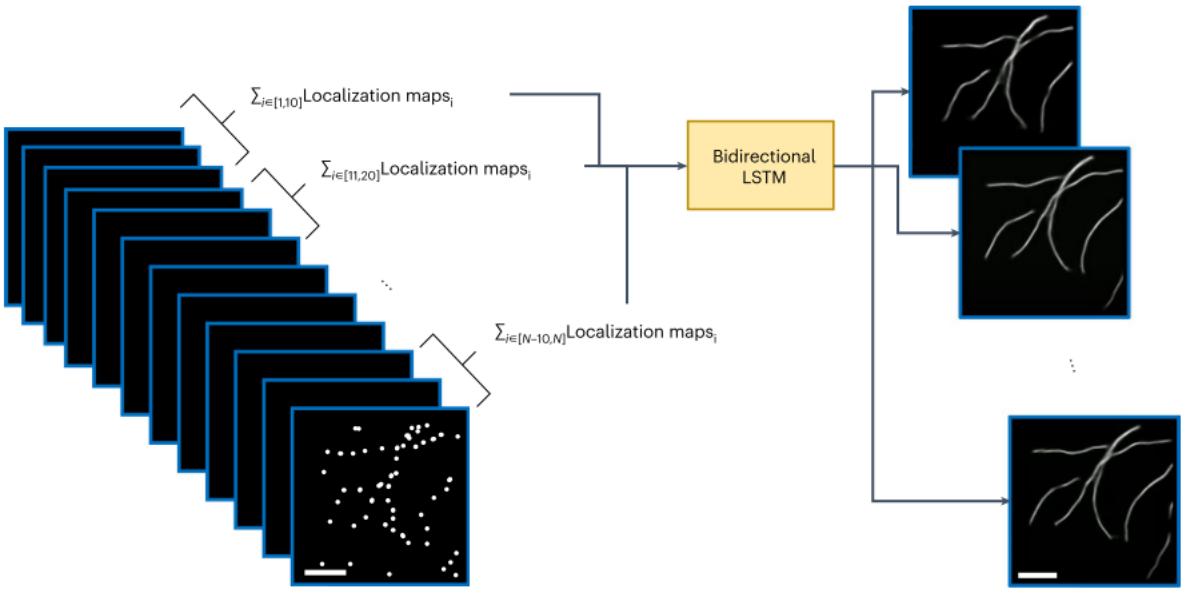


Figure 3-2: DBlink uses as input localization maps, which are localizations summed from N amount of recorded frames and outputs a super-resolved image for each provided localization maps. Source: [8].

3-1-2 DBlink model in ULM

Base DBlink model. The Base DBlink model with unchanged model architecture that is used with our generated renal artery tree dataset is shown in Table 3-1.

Table 3-1: Layer by layer specifications in order of the base DBlink model. (The hidden activation function ReLU in the CNN head is not included in table for simplicity.)

Operation	**In → Out ch.**	**Kernel**	**Padding**	**Stride**	**Input** $(H \times W)$	**RFd**
Input	$1 \to 1$	—	—	—	128×128	1
Forward branch						
ConvLSTM layer 1	$1 \to 4$	5×5	2	1	128×128	5
ConvLSTM layer 2	$4 \to 4$	5×5	2	1	128×128	9
Reverse branch (mirrors forward path)						
ConvLSTM layer 1 rev	$1 \to 4$	5×5	2	1	128×128	5
ConvLSTM layer 2 rev	$4 \to 4$	5×5	2	1	128×128	9
Concatenate (fwd + rev)	$8 \to 8$	—	—	—	128×128	9
Frame-wise CNN head						
Conv2d 1	$8 \to 128$	5×5	2	1	128×128	13
Conv2d 2	$128 \to 256$	5×5	2	1	128×128	17
Conv2d 3	$256 \to 64$	5×5	2	1	128×128	21
Conv2d 4	$64 \to 1$	5×5	2	1	128×128	25

Dilated DBlink model Unlike reconstructing small microtubule structure using base DBlink model whith a small receptive field (RFd), the whole simulated rat renal artery tree structure spans across several hundreds of pixels. By using only base DBlink model, the model may only use small local information to reconstruct the super-resolved image, which may not contain the necessary spatial information needed to construct the structure correctly.
To increase the RFd of the model, the method of repeatingly increasing dilation step in each layer from Yu *et al*[32] was implemented, which allows for exponential RFd size increase. The CNN head of the base DBlink model was modified to include the use of increasing dilations and an extra 2-D convolutional increasing the total RFd to 133 as shown in Table 3-2.

3-2 Data simulation

For the simulation of the dataset that is used for both training and benchmarking of the proposed model, a simulation of microbubble (MB) flowing through a synthetic 2-D renal arterial tree of a rat is generated.

3-2-1 Renal arterial structure

A 2-D renal flat arterial structure is created by following the Strahler ordering rule. Strahler ordering begins by labeling all the vessel segment ends as 0 order. And by following the upstream when 2 vessels of same order joins together at a bifurcation, the order of the upstream

Table 3-2: Layer by layer specifications in order of the Dilated DBlink model with increased receptive field.(The hidden activation function ReLU in the CNN head is not included in table for simplicity.)

Operation	In/Out ch.	Kernel	Padding	Stride	Input ($H\times W$)	RFd
Input	1/3→1	—	—	—	128×128	1
Forward branch						
ConvLSTM 1/3	1/3→4	5×5	2	1	128×128	5
ConvLSTM 2	4→4	5×5	2	1	128×128	9
Reverse branch (mirrors forward path)						
ConvLSTM 1 rev	1/3→4	5×5	2	1	128×128	5
ConvLSTM 2 rev	4→4	5×5	2	1	128×128	9
Concatenate (fwd+rev)	8→8	—	—	—	128×128	9
Dilated–CNN head						
Conv2d 1	8→128	5×5 (dil. 1)	2	1	128×128	13
Conv2d 2	128→256	5×5 (dil. 2)	4	1	128×128	21
Conv2d 3	256→256	5×5 (dil. 4)	8	1	128×128	37
Conv2d 4	256→128	5×5 (dil. 8)	16	1	128×128	69
Conv2d 5	128→64	5×5 (dil. 16)	32	1	128×128	133
Conv2d 6	64→1	1×1	0	1	128×128	133

parent vessel segment is labeled with 1 order higher than its children [9], see Figure 3-3. The Strahler ordering has been shown to correlate well with vessel radius [9]. In Nordsletten *et al.*, the relationship between the Strahler order and the segment lengths and radii of the renal arterial tree is modeled with normal distribution, see Table 3-3.

For simplicity the renal arterial tree contains only bifurcations. The construction starts from a single order 10 root and recursively bifurcates at each segment vessel until order 3 is reached. At every bifurcation split, the length and radius of the two daughter segments are drawn from the corresponding normal distribution $\mathcal{N}(\mu_{L,s}, \sigma^2_{L,s})$ and $\mathcal{N}(\mu_{R,s}, \sigma^2_{R,s})$ with data from Table 3-3. The bifurcation angle to the left or right of each children with respect to the parent vessel segment is sampled from $\mathcal{N}(\mu = \pi/6,\ \sigma = \pi/24)$.

3-2-2 MB propagation simulation

Poiseuille flow. The simulated MB is assumed to be moving with the blood flow. Since blood flow in vessels follows an overall poiseuille law [28], the axial velocity profile flowing in a circular pipe is given by:

$$v_x(r) = \frac{(R^2 - r^2)G}{4\mu} \tag{3-1}$$

where R is the vessel segment radius, r the radial coordinate, G pressure gradient and μ blood viscosity. The axial position of each MB position is updated in each frame by multiplying $v_x(r)$ by the timestep Δt of the ultrasound system.

The maximum axial inlet velocity at the centerline of the root of the renal trees that are simulated is set to have a velocity of $1cm/s$.

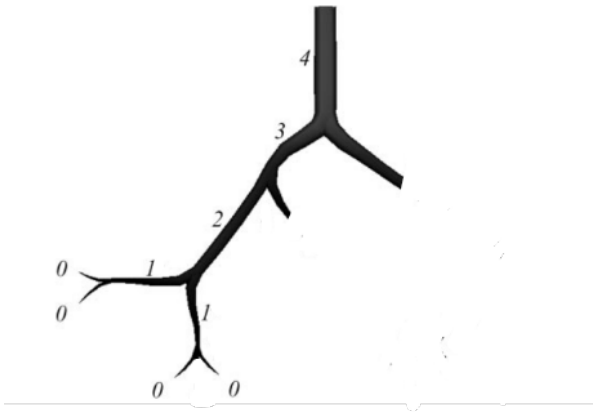


Figure 3-3: Strahler ordering of a vascular tree with bifurcations. Source adapted from [9].

Table 3-3: Post-scaled mean and standard deviation (s.d.) normal distribution values of renal arterial tree vessel segments according to their Strahler ordering. Source adapted from [9].

Order	**Length mean (μm)**	**s.d. (μm)**	**Radius mean (μm)**	**s.d. (μm)**
10	666.67	22.22	216.10	3.16
9	960.00	143.78	191.42	11.86
8	5983.33	295.78	139.83	13.41
7	1677.33	456.22	86.15	16.04
6	687.33	149.78	53.87	8.34
5	340.67	44.44	44.23	6.54
4	667.33	48.00	39.29	0.72
3	437.33	63.56	29.87	0.23
2	269.33	86.67	20.06	4.60
1	282.00	62.89	13.90	2.53
0	208.00	63.33	10.08	0.09

Additionally, the cyclic pulsatility blood flow due to the rithmic contractions of the heart is not considered in the blood flow simulation and is assumed to have a stationary blood flow.

Volumetric flow rate conservation. For conservation of the volumetric flow rate at each vessel segment, an assumption is made that both childs share same gradient pressure G_{child} :

$$Q_{\text{parent}} = R_{\text{p}}^4 \, G_{\text{p}} = (R_1^4 + R_2^4) \, G_{\text{child}}, \tag{3-2}$$

which yields

$$G_{\text{child}} = \frac{R_{\text{p}}^4}{R_1^4 + R_2^4} \, G_{\text{p}} \tag{3-3}$$

for every bifurcation encountered during the traversal of the renal tree. This relation ensures flow rate conservation at each vessel segments without explicitly solving the global pressure network system.

Spawning MBs. At the first frame simulation, the MBs axial coordinate l are sampled from a uniform distribution from the total segment length L of a path starting from the root order 10 Strahler order to the end of order 3 Strahler segment. The radial coordinate r of the MB was sampled from a parabolic distribution that is obtained by normalizing Eq. (3-1). After a MB has left the end of its path, a new MB is spawned with its axial coordinate set to 0 and a new radial coordinate being sampled from the parabolic distribution.

Path selection. When a MB is at bifurcation, the probability of going to one of the two possible child vessel segment choices is given by :

$$p_i = \frac{R_i^2}{R_1^2 + R_2^2}, \qquad i \in \{1, 2\}. \tag{3-4}$$

3-2-3 Frame and dataset simulation

Frame simulation. For the generation of the summed localized frames known as localization maps that are used as input for DBlink model in ULM, it was assumed that all MBs during the simulation can be localized with a precision of $\frac{\lambda}{10}$ and hence same value as the isotropic spatial pixel resolution size (see Eq. (2-1) for calculation of wavelength λ). For the generation of the velocimetry tracked maps, it was assumed that all individual MBs can be tracked at all time during the simulation.
The parameters used for the simulation are shown in Table 3-4.

Table 3-4: Parameters used for each data simulation run.

Parameter (symbol)	Value
Ultrasound center frequency, f	7.6 MHz
MB per simulation	100
Frame rate frequency	100 Hz
Frames per simulation	1500
Acquisition time	15 s
Blood viscosity, μ	3.27×10^{-3} Pa s
Speed of sound in blood	$1540 m/s$

Dataset generation. A consecutive block of $N_{sum} = 30$ localization frames are collapsed into a single localization map, by counting each time a MB is localized in a pixel, the pixel value is incremented by 1.
For the velocimetry map, every MB track that intersects a pixel during the N_{sum} frames, the pixel value is the velocity of the MB in cm/s. The velocimetry map consists of two channel, with a channel each for velocity components along the 2-D grid axes. The velocimetry map is also used as the output of the conventional ULM method which the proposed models will be benchmarked against.
The binary ground-truth mask of the renal artery structure has the same resolution as the localization frames and the pixel values are set to binary value 1 where vessel structure is present and 0 otherwise, see Figure 3-4 for full Field of View (FOV) of the renal tree structure.

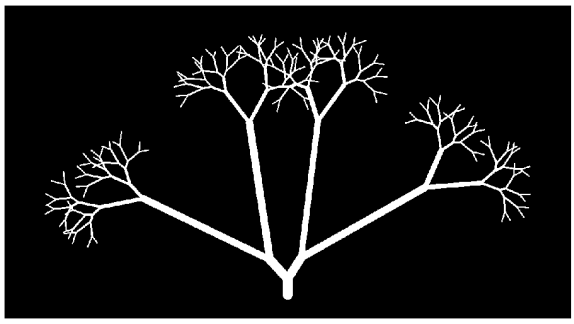


Figure 3-4: Full FOV of a simulated rat renal artery tree vessel structure.

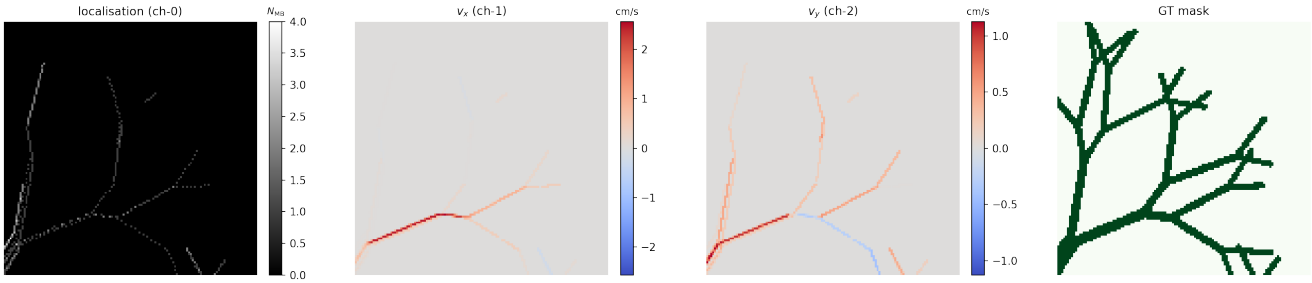


Figure 3-5: Simulated rat renal artery training data of single frame composed of different channels along with the ground-truth binary mask. From left to right, localization map, velocimetry map v_x and v_y and ground-truth binary mask

Because of GPU-memory limitation, the training set is stored as 128×128 patches that are uniformly sampled from the full FOV image and contains at least 5 % of pixels of vessel structure pixels. The resulting training set dimensions is then given by:

$$X_{train} = [N, T, C, H, W] = [1116,\ 50,\ C,\ 128,\ 128],$$

see Figure 3-5. The validation and test sets keep the full FOV of the whole renal tree structure:

$$X_{val} = [6,\ 50,\ C,\ 655,\ 1185] \quad , \qquad X_{test} = [10,\ 50,\ C,\ 655,\ 1185].$$

Here N is the number of samples, T the number of summed frames, C the channel count, and H, W the patch height and width.

Types of inputs for training. To answer subquestion on how different inputs available to ULM affects the model performance, there will be two types of data used for training. The first is the base input, which consists of a single channel containing data of only localization maps as mentioned previously. The localization maps not only provide structure information, but also the amount MB that has passed through a vessel segment of a given radius. From Eq. (2-11) and Eq. (2-10), there is a relation between the amount of MB passed through a vessel of a given radius. With this relation, it is expected that the model is able to reconstruct the super-resolved image of the renal tree with low hallucination rate.

The other input which will be used on training contains three channels, which contains one channel of localization maps and two other channels of velocity tracks each with velocity components along the 2-D axes. Since there is a relation between MB velocity and vessel radius as seen on Eq. (3-1), it is also expected that the addition of velocity channels will also improve the performance of the model in reconstructing the super-resolved renal tree.

3-3 Experiment setup and measuring model performance

To measure the performance of the model and answering the research questions, the task in this research is framed as a binary semantic segmentation task. The subsections below describe the loss functions used to train the proposed models, the performance metrics to quantify the performance of the model and finally the procedure for the hyperparameter optimization of the model.

3-3-1 Loss functions

The loss functions used for training of the proposed models consist of both regression and segmentation loss functions. For regression loss functions, the mean squared error (MSE) and total variance (TV) losses are used. Since the task in this thesis is a segmentation task, binary cross-entropy (BCE) is used as a segmentation loss.

Mean squared Error. The MSE loss is the average squared pixel value difference between the target and prediction:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{B\,H\,W} \sum_{n=1}^{B} \sum_{i=1}^{H} \sum_{j=1}^{W} (\hat{y}_{n,i,j} - y_{n,i,j})^2. \tag{3-5}$$

Where B is the batch size, H, W the height and width of the image, and $\hat{y}$ and y the prediction and binary ground-truth mask.

TV loss. The TV loss is the average squared error between neighboring pixels. This loss is used in combination with other loss functions such as the MSE. The loss also acts as a regularizer and encourages model to predict a smooth spatial structure [8]:

$$\mathcal{L}_{\text{TV}} = \lambda_{\text{TV}} \frac{2}{B\,H\,W} \sum_{n=1}^{B} \sum_{i=1}^{H-1} \sum_{j=1}^{W-1} \Big[(\hat{y}_{n,i+1,j} - \hat{y}_{n,i,j})^2 + (\hat{y}_{n,i,j+1} - \hat{y}_{n,i,j})^2 \Big], \tag{3-6}$$

with λ_{TV} the TV weighting.

BCE. The BCE loss measures how far each predicted pixel probability is from the binary ground-truth pixel label. Minimizing this loss results in the model have a higher probability that the pixels are labeled correctly. The BCE equation is given by [11]:

$$\mathcal{L}_{\text{BCE}} \;=\; -\frac{1}{B\,H\,W}\sum_{n=1}^{B}\sum_{i=1}^{H}\sum_{j=1}^{W}\Big[y_{n,i,j}\;\log\hat{y}_{n,i,j}\;+\;(1-y_{n,i,j})\;\log(1-\hat{y}_{n,i,j})\Big], \tag{3-7}$$

3-3-2 Performance metrics

To quantify the performance of the model in the binary segmentation task for both validation during training or on unseen test data for evaluation. The metrics used here are based on confusion matrix. The confusion matrix is shown in Table 3-5.

Table 3-5: Confusion matrix for binary vessel segmentation.

Ground-truth	**Predicted**	
	Vessel	Background
Vessel	True Positive (TP)	False Negative (FN)
Background	False Positive (FP)	True Negative (TN)

Recall. Recall measure the TP recovery rate and is given by [33]:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3-8}$$

Recall is used as an evaluation metric to measure the saturation percentage of predicted vessels as a function of acquisition time.

Precision. Since Recall does not account for FP, the precision metric is used as an evaluation metric for hallucination of fake predicted vessels FP. The precision is given by:

$$\text{Precsion} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3-9}$$

F1-score. The F1-score, also known as Sørensen-Dice index is defined as the harmonic mean between the recall and precision.[33]. The Precision metric is used in combination with Recall to balance the FP and FN errors to form the F1-score. The F1-score

The F1-score is well suited for tasks where there is an imbalance of class labels since it ignores the TN which can skew the performance of the model. In our case, the vessel pixels account for roughly 5 % of the total image pixels. The F1-score will be used for both validation and evaluation of the model. By using F1-score as validation metric we ensure that the both recall and precision are

The equation for F1-score is given by:

$$\text{F1-score} = \frac{2}{\text{Recall} + \text{Precision}} = \frac{2\text{TP}}{\text{TP}+\text{ FP } + \text{FN}} \tag{3-10}$$

3-3-3 Training and model validation step

Training step. At every optimisation step a minibatch:

$$X_{train} = [B,\ 1{:}T_{\text{sampled}},\ C,\ 128,\ 128]$$

is drawn, where B is the batch size.
The sequence length T_{sampled} is sampled from a discrete uniform distribution in range of [1,50]. Using this randomly chosen sequences encourages the network to predict the binary ground-truth mask of the renal artery structure from the limited amount of acquisition time. The corresponding loss is then computed with one of the loss functions from Section 3-3-1 ($\mathcal{L}_{\text{BCE}}$ or $\mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{MSE}}$).

Validation step. After each epoch, the model is validated at six acquisition times:

$$T_{val} = 1,\ 10,\ 20,\ 30,\ 40,\ 50.$$

Resulting in the following validation data:

$$X_{val} = [B,\ 1{:}T_{val},\ C,\ 655,\ 1185].$$

By validating only these six T_{val} sequences, the time spent during each validation step is maintained low.

3-3-4 Hyperparameter optimization

To optimize the hyperparameters, the hyperparameter optimization framework Optuna[34] is utilized. In each Optuna study, consisting of several number of trials where in each trial, Optuna uses a sampler algorithm to adaptively choose parameters values or choices which are associated with best model performance. The default sampler algorithm Tree-structured Parzen Estimator (TPE) is used.

For each studies that are conducted, the following training hyperparameters are fixed:

- **Optimizer**: Adam, $\beta_1 = 0.99,\ \beta_2 = 0.999$
- **Learning rate scheduler**: With a patience of 3 epochs, the learning rate is decreased by scale of 0.1 if the model performance has not improved.
- **Trial length**: Each Optuna trial lasts for 20 epochs.
- **Trials per study**: Each Optuna study consists of 20 trials.
- **Median pruner warmup**: Each trial will be pruned whenever the model performance in each epoch is worse than the median of other trials in the same study and same epoch. The median pruning is activated only after 5 trials has been conducted in each study and only after 10 epochs.

After finding the hyperparameters from each of the studies, the models are then retrained with the found hyperparameters for 100 epochs and with early stopping condition of 10 epochs to ensure optimal model weights.

Hyperparameter search space. The hyperparameters which are optimized with Optuna are shown in Table 3-6.

Table 3-6: Optuna hyperparameter search space.

Hyper-parameter	**[Range] / { choices}**
Learning rate	Log-uniform sample of $[2 \times 10^{-4}, 2 \times 10^{-3}]$
Batch size	$\{1, 4\}$
Loss functions	$\{ \mathcal{L}_{\text{BCE}} , \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{MSE}} \}$

Study variants. Three different studies will be conducted in order to answer the first research sub-questions. To understand whether the size of the simulated renal tree structure affects the performance of the model, two different studies S-1 and S-2 are performed with each having a different models with different RFd sizes as seen on subsection 3-1-2 and same one channel input type of localization map.
To answer whether different input type as mentioned in subsection 3-2-3, an additional study S-3 is done with the best performing model from S-1 versus S-2. In S-3 a different input type is used, which is three channels consisting of both the localization map and the velocity track channels. In Table 3-7 a short summary of the different studies are provided.

Table 3-7: Configuration of the three Optuna studies.

Study	**model**	**Input types(Channels)**
S-1	Base DBlink	Localization maps (CH1)
S-2	Dilated DBlink	Localization maps (CH1)
S-3	Base/ Dilated DBlink	Localization maps + velocity tracks (CH3)

Chapter 4

Results

In this chapter the results from evaluating the performance of the trained models and their reconstruction of the rat renal artery are shown. Section 4-1 shows the trained models validation performance and their hyperparameters. Section 4-2 shows the different trained models evaluation compared to each other at different region of interest (ROI) and Section 4-3 the reconstruction of renal artery using the best model compared against the conventional Ultrasound localization microscopy (ULM) method. Finally, in Section 4-4 the evaluation of the models at different microbubble (MB) concentrations.

4-1 Trained models

From the three different studies using Optuna hypertuning, the models with the highest validation score is then trained for 50 additional epochs with a stop condition of 10 epochs for optimized model weights.
In Table 4-1, a summary results of the optimized hyperparameters along with their validation scores, which shows that the dilated DBlink model trained with three input channels using both localization maps and velocimetry tracks achieves the highest F1-score of 0.886 on the validation set. The dilated DBlink achieved a higher F1-score of 0.877 than the base Dblink while using same Localization map input.

4-2 Performance evaluation

The reconstruction performance of the renal tree using the trained models is evaluated with F1-score, recall, and precision. The metrics are computed on both the full Field of View (FOV) image and on the ROIs highlighting regions with either sparse large radius vessels or dense small radius vessels, as seen in Figure 3-4.

Table 4-1: Optimized models resulting from the Optuna studies along with the hyperparameter values and validation score.

	S-1	**S-2**	**S-3**
Best F1-score	0.869	0.877	0.886
Model	Base DBlink	Dilated DBlink	Dilated DBlink
Input channels	1	1	3
Loss function	BCE	BCE	BCE
Batch size	1	1	1
Learning rate	0.00045	0.00173	0.00115

4-2-1 Full FOV performance

For the full FOV reconstructions, Figure 4-1 shows F1-score, precision, and recall as a function of acquisition time (image examples are in Figure A-5). Since the velocity tracks are used as Conventional ULM output, the nonzero velocity tracks are set as binary values 1 and 0 otherwise.
The plot labels indicates the input channels and whether the model is the base DBlink model or dilated. As seen in Figure 4-1, both precision and recall increases with acquisition time for all the models. From the precision graph at acquisition time T= 1 CH3-dilated has 0.024 and 0.028 increased precision over CH1-dilated and CH1-base model respectively and also maintains the precision advantage as acquisition time increases. In the recall graph at acquisiton time T= 1, CH1-base has recall of 0.47 while dilated CH3 and CH1 has recalls of 0.54 and 0.53 respectively. The CH3-dilated model consistently outperforms the others in F1-score at all acquisition times, due to maintaining both highest recall and precision at all times compared to other models.

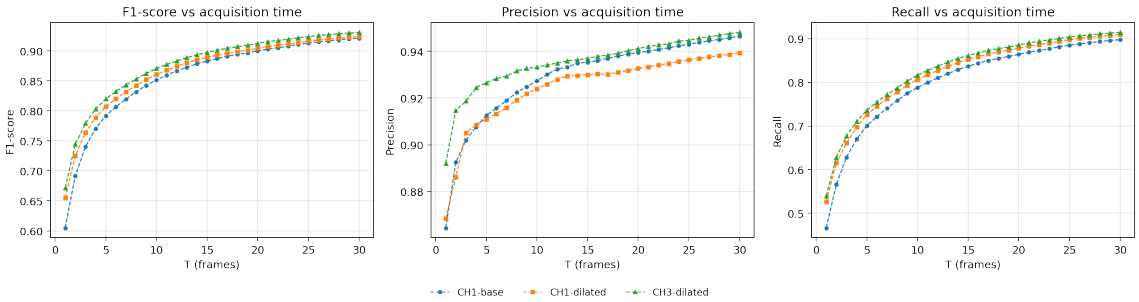


Figure 4-1: F1-score, precision and recall evaluations of the models on test data as a function of acquisition time T, where each T value represents a single frame of Localization map (or velocimetry track frame). The labels are named in term of its channel input along with dilated or base DBlink model.

4-2-2 Performance on ROI with different vessel radii

Performance is also evaluated on two ROIs: one containing dense small radius vessels (Strahler orders 7–3) as seen on Figure 4-3 and one containing sparse large-radius vessels (Strahler or-

ders 10–8) as seen on Figure 4-4. (For the ROI's in the test samples, see Figure A-6.)

From the reconstruction of dense small radii vessels on Figure 4-3, the hallucination of vessels or false positives has been identified to come in three types:(i) overestimation of radius size by having false positives surrounding both sides of a vessel, (ii) fusing of non-overlapping vessels that are few pixels apart (see blue circle in Figure 4-3) and (iii) one side of the vessel to have false positives when only a single MB flows through a vessel segment and is flowing close to the wall of the vessel. For sparse and large vessels on Figure 4-4, type (i) and type (iii) false positives mainly appears.

The dilated models reconstructs sparse large vessels when compared to the CH1-base model when looking at T= 1 recalls with dilated CH3 and CH1 having 0.83 and 0.79 respectively while CH1-base only has 0.67 recall. This can also be seen directly in Figure 4-4 at acquisition time T=1 where CH1-model struggles to recognize MB belonging to the same vessels when looking at the historical MB tracks from conventional ULM method.
The small vessels needs more acquisition time to reconstruct when comparing their recalls to the large vessels counterpart. This can be seen in Figure 4-3 with needing more acquisition time for each of the small vessels to have at least a MB to flow through them.

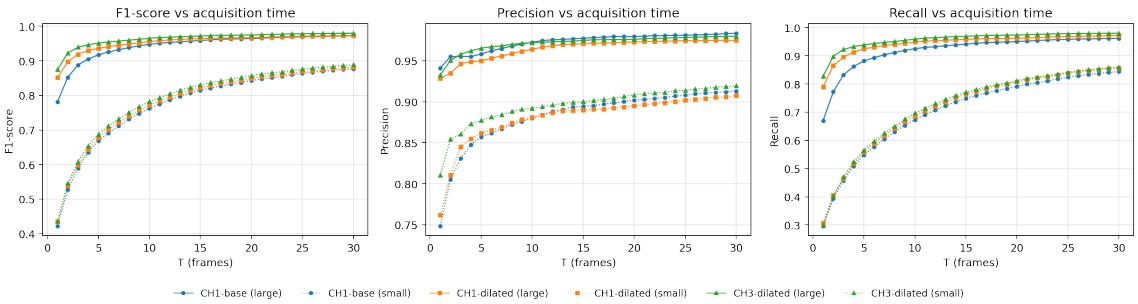


Figure 4-2: F1-score, precision and recall evaluations of the models on test data as a function of acquisition time T for ROI consisting of sparse large radii vessels and ROI consisting of dense small radii vessels.

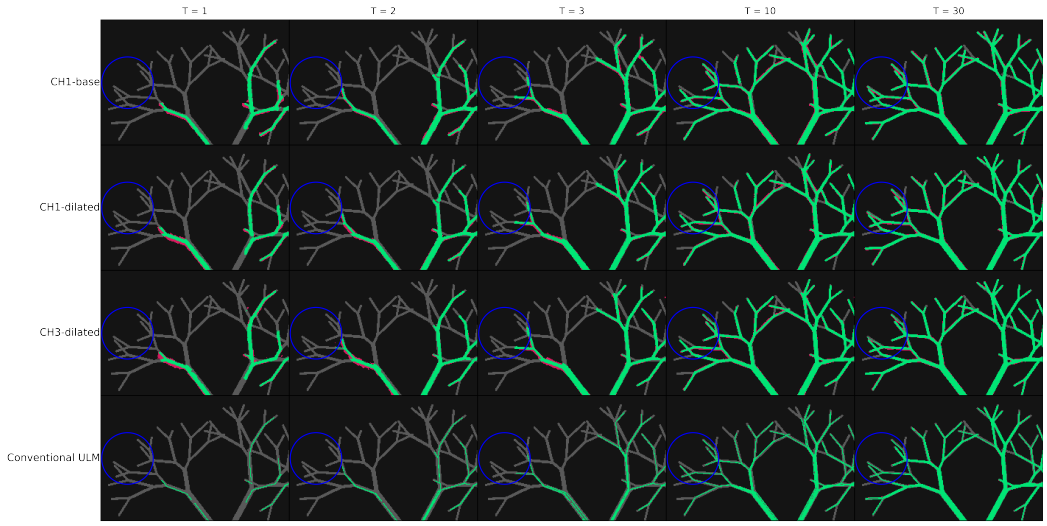


Figure 4-3: Reconstruction of ROI containing small radii vessels at different acquisition times T. With green indicating true positives, magenta indicates false positives and grey false negatives. Blue circle shows very close non-overlapping vessels which are fused together from hallucination.

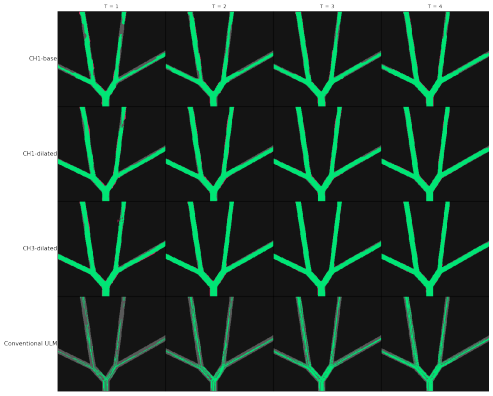


Figure 4-4: Reconstruction of ROI containing sparse large radii vessels at different acquisition times T.

4-3 Benchmarking

The conventional ULM method which uses accumulation of MB velocity tracks to reconstruct the rat renal artery is used as baseline for benchmark to the CH3-dilated model. In Figure 4-5 shows that the CH3-dilated model has a recall difference at T=1 of 0.4 higher recall while having only 0.11 lower precision only compared to the conventional ULM method.

In Figure 4-6 the recall as a function of time T for each Strahler Order is compared, it shows that the CH3-dilated model always has higher recall for all Strahler orders at same acquisition times indicating faster reconstruction times for similar sized radii vessels when comparing to the conventional ULM method. The starting recall for the CH3-dilated model starts at close to 1 recall when acquisition time T= 1 and decreases as the Strahler order is reduced.
The time gap ΔT, is the time which the conventional ULM method reaches the same recall score as the CH3-dilated at acquisition time T= 1. This value can be interpreted as time

saved, which also decreases as vessel order decreases as the small vessels need to wait for an event where at least a single MB to flow through it to reconstruct the vessel.

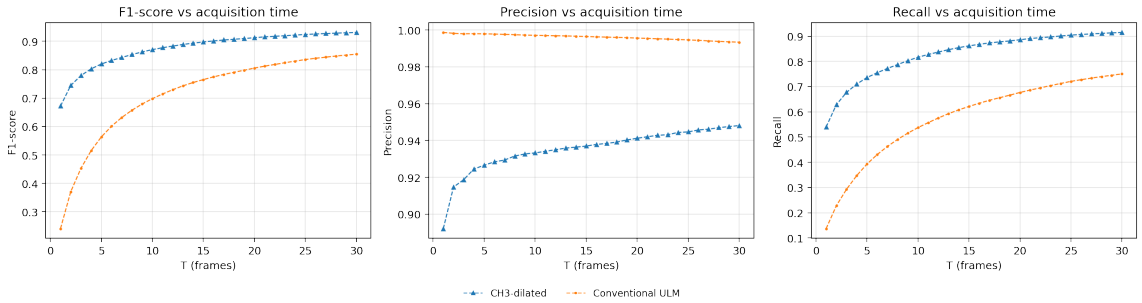


Figure 4-5: F1-score, precision and recall evaluations of the models on test data as a function of acquisition time T for full FOV image. Comparison of model CH3-dilated versus Conventional ULM method.

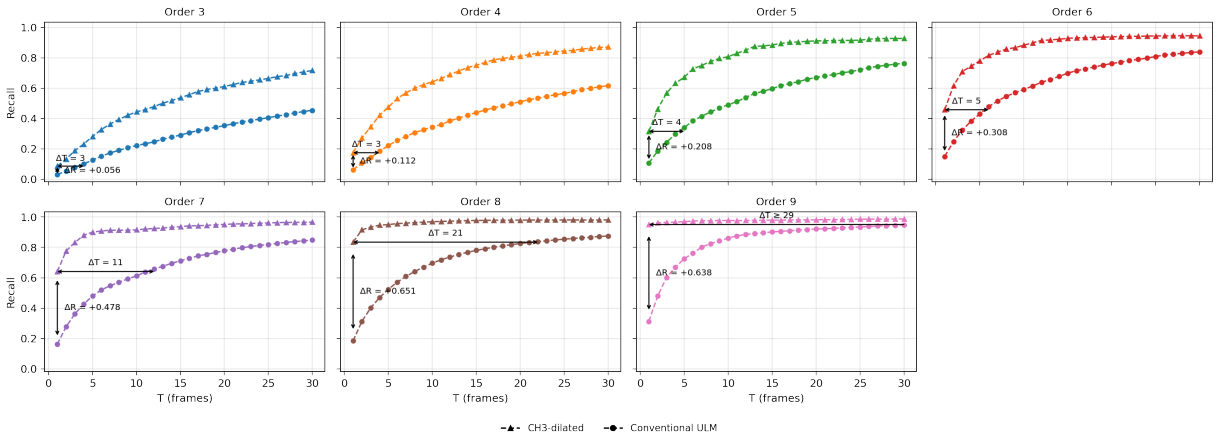


Figure 4-6: Per Strahler Order recall as function of time T graph comparison between CH3-dilated model and Conventional ULM. With arrows showing the ΔT time required to reach the same recall of CH3-dilated model and ΔR the recall difference at T$=1$.

4-4 Performance in varying MB concentration

The CH1- and CH3-dilated models are evaluated and compared using test sets with different MB concentrations than those used during training, namely 50 and 150 MBs. Figure 4-7 shows that precision decreases as MB concentration increases when comparing at the same acquisition time. The maximum precision drop for the CH1-dilated model is about twice as large as that of the CH3-dilated model. In Figure 4-8, the largest vessels in the dense small-vessel ROI have their radii overestimated at higher MB concentrations when compared with reconstruction of the same model, this explains the observed precision drop seen on Figure 4-7.

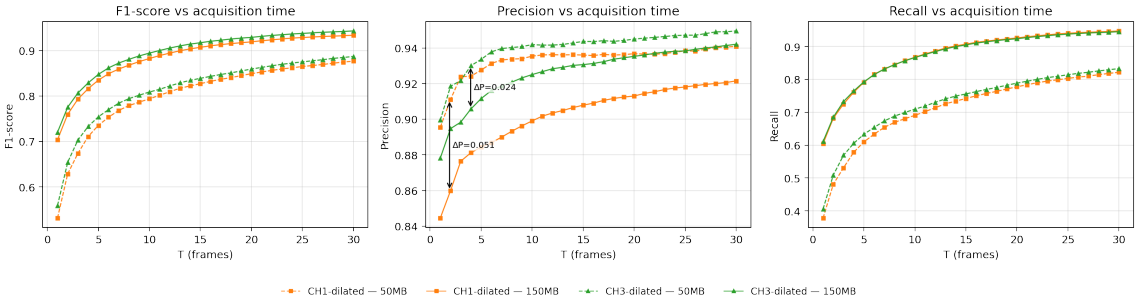


Figure 4-7: F1-score, precision and recall as a function of acquisition time T for full FOV image using the dilated models test data with varying MB concentrations.

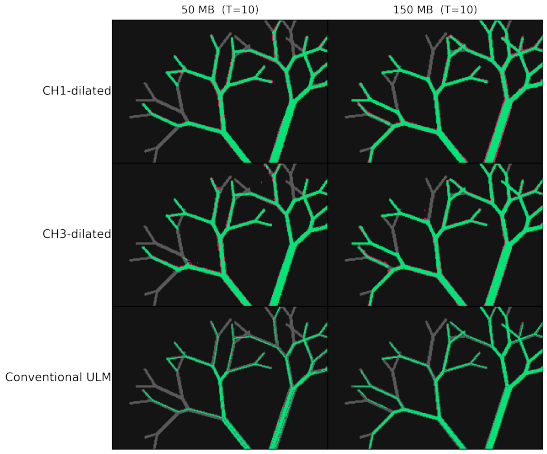


Figure 4-8: Reconstructed image comparison of small vessel ROI at T = 10 at different MB concentrations.

Chapter 5

Discussion

In this chapter, the results from Chapter 4 and related topics are discussed. Sections 5-1 and 5-2 discuss the effects of receptive field size and input type on model performance. Section 5-3 the comparison between the best trained models and the baseline conventional Ultrasound localization microscopy (ULM) are discussed method and outlines the reduction of acquisition time. Finally, the methodology regarding the simulation of the training data is discussed.

5-1 Effect of model receptive field size

When comparing reconstructions of large, sparse vessels using the CH1-base and CH1-dilated models (with both trained on the same localization map input), CH1-dilated recall advantage of 0.12 recall at the low acquisition time T= 1. This can be mainly attributed to its larger receptive field, which provides a greater spatial context. With the larger receptive field (RFd), the model can recognize microbubble (MB) paths belonging to the same vessel segment, enabling the model to predict more positives in the prediction when a vessel is long. In contrast, the CH1-base model, with its smaller receptive field, lacks the spatial context to associate the MB paths that are far apart in long vessels. As a result, the model makes less vessel predictions leading to less recall when compared to CH1-dilated.
To apply the DBlink model in vasculature in other organs, the receptive field size of the model should be appropriately adjusted for the size of the vessel structure of interest to increase the recall performance.

5-2 Effect of different types ULM inputs

When reconstructing the rat renal tree and comparing same dilated DBlink models with input of models using 1 channel localization maps or 3 channel localization maps with addition of MB velocity tracks. The CH3-dilated model achieved a similar recall score at early acquisition times and a higher precision. The precision increase can be attributed from the

added velocity track channels to the input, which provides richer information to associate MB paths within the same vessel segment. For example, MBs traveling close together in a given segment and direction exhibit similar velocity components, allowing the model to make more true positive predictions which was observed from the reconstruction of the large vessels.

Using tracked MBs in addition to the localization map also makes CH3-dilated more stable precision across different MB concentrations as the result showed, compared to using localization map alone. This was observed when evaluating both models on test data simulated at 50 and 150 MB condition with the CH1-dilated precision error being twice of the CH3-dilated model.

For future research it may be worth looking into using low resolution ultrasound frames as input to study how it compares to localization maps or velocity tracks, as this may illustrate the best input type to consider for the deep learning model in the reconstruction of a ULM super resolution image. As the current deep learning models in ULM which learns to predict the super resolution reconstructions such as Chen *et al.* [17] primarily uses as input these low resolution ultrasound frames.

5-3 Reducing the acquisition time with DBlink

In the benchmark, the best F1-score model CH3-dilated was evaluated against the baseline conventional ULM method. When looking at the global performance or reconstruction of the full Field of View (FOV) of the rat renal artery tree, the CH3-model was found to have a recall advantage of 0.4 at acquisition T= 1. This implies a lower acquisition time, so fewer frames are needed to achieve the same reconstruction as the baseline.

When looking at the recall per Strahler order, high Strahler order (large radii) vessels tends to start with a high recall score and a long time needed for baseline method to reach this same recall resulting and hence less frames needed to reach the same reconstruction. For a Strahler order 9 vessel the time gap $\Delta T = 29$ which corresponds to ≈ 8.7 seconds time reduction in the simulation when compared to conventional ULM.
It was also shown that as the Strahler order decreases (corresponding with smaller radii vessels), the time gap ΔT to reach same recall also diminishes as the vessels needs at least a single MB to flow through them. If the primary goal of the reconstructions of the smallest radii vessels, this deep learning method may not be suited as the acquisition reduction is minimal.

5-4 Simulated data

Although the models achieve a higher F1-score than conventional ULM on the simulated rat renal artery dataset, this may not translate to application in real ULM due to the assumptions and simplifications made in simulating the training data.
Several simplifications were made in the simulation. For example, only Poiseuille flow was used to model blood flow, while higher-order effects were omitted. The cyclic pulsatile nature

of blood flow was also not included. Additionally, the simulated renal arterial tree was heavily simplified and contains only bifurcations and is modeled as a flat 2-D structure, which also neglects out of plane flow.

To address the problem of the unrealistic simulated rat renal tree structure and the blood flow simulation within it, it is recommended to use physically realistic simulation frame work from Blanken *et al.*[35]. Their framework uses a 3-D mesh structure of a real rat renal artery and simulates the blood flows according to the more complex Navier Stokes equation and they were also able to simulate pulsatile bloodflow. Given the limited availability of high-quality 3-D meshes of rat renal arteries and the considerable computational cost for simulation of the blood flow, additional time and resources are necessary to curate a realistic simulation data for deep learning.

Furthermore, it was assumed that every localizations were assumed to have precision of $\lambda/10$ and every MB can be tracked. In reality, not every MB can be localized with the same precision due to multiple factors, such as non-homogeneous speed of sound of the tissues and the addition of noise in the Radio Frequency (RF) signals. Similarly not every MB can be tracked due to missing MB localizations. Further research is needed to see how the model generalizes in these differing MB localization precision and imperfect MB tracks.

Chapter 6

Conclusion & Recommendations

6-1 Conclusions

The goal of this research was to explore how the DBlink deep learning model can be trained for Ultrasound localization microscopy (ULM) and how it can reduce acquisition time when compared to the conventional ULM method. In this chapter the research questions will be answered with the results that were obtained from the research.

How can the DBlink model be trained for ULM?

A simulated training data was made in silico by first creating a 2-dimensional geometry structure of a rat renal arterial tree consisting of only bifurcations by using data from Strahler ordering [9]. Within the structure microbubble (MB)s were propagated by assuming simple poisseuille flow. Two different types of input training sets were simulated to test how these affects the performance of the model. One training set consists of a single channel of a frames of summed MB localizations named as localization maps. With the other training set containing three channel that consists of a single channel localization map with additionally two channels containing the velocity tracks of the MB.

The training sets were used to train a deeplearning model DBlink [8] which the network architecture consists of a Bi-directional long short-term memory (LSTM) connected with a Convolutional Neural Networks (CNN) head. The receptive field (RFd) size of the DBlink model was increased with the use of dilated convolutions to study how the affect of RFd affects the performance model when reconstructing the rat renal artery tree and compared with the unmodified base DBlink model.

When trained with three channel input type, it was found that the model is more robust in precision performance. The precision drop when reconstructing the renal tree using single channel input was found to be twice as much as the three channel input when evaluated on simulated test data with 50 % less MB and 50 % more MB than the MBs in the training data. It was also found that the three channel input had 2.4 % more precision at low acquisition T= 1 when reconstructing the full Field of View (FOV) of the renal tree over the 1 channel input.

When evaluating the models with different RFd size and trained using only the single channel localization maps, the model with the increased RFd was found to have a higher recall. When reconstructing the sparse large vessel at early acquisition time T= 1, the model with increased RFd had 12 % recall higher than the unmodified DBlink model. The model with higher RFd and thus bigger spatial context was able to recognize flowing MB paths that were far apart in long vessels when there is low amount of MBs flowing through that vessel.

Does the trained DBlink model reduce acquisition time when compared to the conventional ULM method?

The best performing model in terms of F1-score, CH3-dilated (model with increased RFd size and using three channel input) was compared to the conventional ULM. In the comparison, the CH3-dilated model was able to reduce the acquisition time of 29 frames corresponding to ≈8.7 seconds for large radii vessels by having a large recall advantage of 0.64 recall at acquisition time T= 1 over the comparison for Strahler order 9 which corresponds to large radii vessels. This recall advantage diminishes as the vessel radii size decreases as the small vessels are still limited by the long acquisition times for at least a single MB to pass through it.

6-2 Limitations and future research recommendations

For future research, some recommendations are made based on the current results from this research. Due to the various simplifications made to to be able to simulate the training data, it is recommended for the use of a more realistic simulated training data that better reflects the conditions in real life in order for the model to generalize well for real life ULM applications. The realistic training datas could be obtained by using the physically realistic simulation framework from Blanken *et al.* [35].

In this research the input type considered was only the MB localization maps and velocity tracks as input for the deep learning model. In other deep learning model which were applied in ULM for reconstructing super resolution images (such as Chen *et al.*[17]) primarily uses low resolution ultrasound frames as input. By considering these input modalities and comparing their effects on their deep learning super resolution reconstruction, which could potentially establish a new standard input type for the deep learning models in ULM.

Appendix A

Appendix A

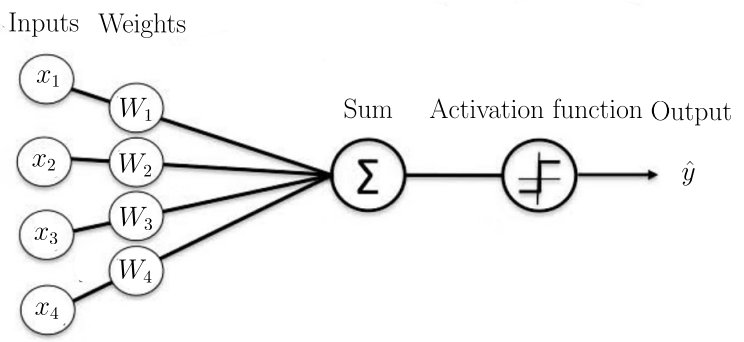


Figure A-1: Graphical illustration of a perceptron. The inputs x_i are multiplied by the weights and summed along with the bias, after which it passes through an activation function to obtain the output. Source adapted from: [10].

A-1 Deep learning basics

A-1-1 Perceptron

The simplest neural network algorithm is the perceptron, which is graphically shown on Figure A-1. This perceptron is a feedforward network, since the input data passes through the model in one direction. The input $\boldsymbol{x}$ is multiplied by a weight and summed together with a bias and the output is then obtained by passing the sum through a nonlinear activation function, which allows a nonlinear function to be modelled.

The output function for the perceptron can be written as:

$$\hat{y} = g(\boldsymbol{W}^T \boldsymbol{x} + c), \tag{A-1}$$

where g is the activation function, $\boldsymbol{W}$ is a vector which contains the weight , $\boldsymbol{x}$ is the input vector containing the input scalars and finally c the bias.

A-1-2 Multilayer perceptrons

To allow for more complex nonlinear functions to be modelled, the perceptrons can be extended to a multilayer perceptrons (MLPs). The MLPs forms the foundation in deep learning models.

The perceptron in Figure A-1 is a single layer perceptron, since the input passes through one layer of perceptron to obtain the output. In MLPs, there can be multiple layers, in which the output of a perceptron is fed as an input for the subsequent perceptrons in series. The last layer of perceptrons is the output layer and layer of perceptrons between the model inputs and the output layer are named the hidden layers. The number of outputs in each layer is the width of the network.

The simplest MLPs are the single layer neural network, which is shown in Figure A-2 where the summation node and the activation function is combined to one node known as a unit and the network has a single hidden layer [11]. The output function for this network is then :

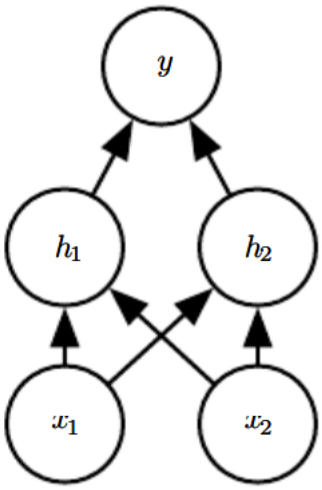


Figure A-2: A simple single layer Neural network with 1 hidden layer and 1 output. Source: [11]

$$\boldsymbol{h} = g(\boldsymbol{W_1}^T \boldsymbol{x} + \boldsymbol{c_1}), \tag{A-2}$$
$$\hat{\boldsymbol{y}} = g(\boldsymbol{W_2}^T \boldsymbol{h} + \boldsymbol{c_2}), \tag{A-3}$$

where $\boldsymbol{W_i}$ are the weight vectors before each unit layers in order and $\boldsymbol{h}$ is the output vector of the hidden layer and $\boldsymbol{c_i}$ are the bias vectors. In Figure A-2 the variables are scalar, but can be combined to vector notations as in Eq. (A-3) and Eq. (A-3).

A-1-3 Activation function

To complete the MLPs model, a specific activation function must be chosen for the hidden and output layers depending on the type of task. Every activation function can be used in both the hidden and the output layer, but to train a good model it is necessary to know where and when to apply certain activation functions. In Table A-1, some common activation functions are listed with its function charecteristics and which part of the layer it is commonly used in.

Some of the listed activations can suffer from vanishing gradient and dead neurons. In vanishing gradient, the gradient of the model decreases due to a combination of the number of hidden layers and the derivatives of the activation. Since the error gradient from the output to the input is calculated using chain rule and the derivatives of activation function being smaller than 1, this means that the gradient approaches to 0 by adding more hidden layers.

The vanishing gradient problem can be solved by using the ReLU function, which always has a constant gradient value of 1 for positive input values. However the ReLU function also has a drawback, the gradient of the function can be 0 for negative values, which causes the weights not to be updated and this is known as the dead neuron. For dead neuron, a leaky ReLU can be used which has a nonzero gradient when it is has negative input.

Table of common activation functions in deep learning			
Name	**Activation function** $g(x)$	**Common uses**	**Characteristics**
Sigmoid	$g(x) = \frac{1}{1+e^{-x}}$	Output: logistic regression, binary classification	Can suffer from vanishing gradient problem if used for hidden layers
Tanh	$g(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$	Output: classification	Can suffer from vanishing gradient problem if used for hidden layers
ReLU	$g(x) = max(0, x)$	Hidden layer function	Standard hidden layer activation function, fast computation, can have dead neurons
Leaky ReLU	$g(x) = \begin{cases} x \text{ if } x \geq 0 \\ 0.01x \text{ if } x < 0 \end{cases}$	Hidden layer function	Similar to ReLU and has no dead neurons problem
Swish	$g(x) = \frac{x}{1+e^{-x}}$	Hidden layer function	Similar to sigmoid function, but without vanishing gradient problem
Softmax	$g(x)_i = \frac{e(x_i)}{\sum_j e^{x_j}}$	Output: Multiclass classification	Mostly used in output layers

Table A-1: Table of common activation functions and their uses in deep learning. Source adapted from: [13].

Training the MLPs

The MLPs is trained by minimizing the loss function, the loss function measures how well the network model performs during its training and during the training this loss is minimized. Minimizing this loss function with respect to the weights will result in the trained model to make better predictions for the given task.

The general loss function used in deep learning is given by:

$$J(\boldsymbol{W}) = \frac{1}{m}\sum_{i=1}^{m} L(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)}, \boldsymbol{W})), \tag{A-4}$$

where $\boldsymbol{W}$ are the weights of the network, L is the per-example loss and lastly $\boldsymbol{x}^{(i)}$, $\boldsymbol{y}^{(i)}$ is the input and output of the $i'th$ example and m is the amount of examples used for training.

The per-example loss depends on the type of task the network has to perform. A typical per-example loss function for regression tasks is the mean squared error loss:

$$L(\boldsymbol{W}) = \frac{1}{2}||\boldsymbol{y}^{(i)} - \hat{\boldsymbol{y}}^{(i)}(\boldsymbol{x}; \boldsymbol{W})||^2, \tag{A-5}$$

where $\hat{\boldsymbol{y}}$ the network output prediction.

A typical loss function for classification task is the cross-entropy:

$$L(\boldsymbol{W}) = -\sum_{j=1}^{k} \boldsymbol{y}^{(j)(i)} log(\hat{\boldsymbol{y}}^{(j)(i)}(\boldsymbol{x}, \boldsymbol{W})), \tag{A-6}$$

where superscripts j and i denotes the category class and the $i'th$ example.

The loss function is minimized by using gradient descent and updating the weights for each unit in the network. The gradient with respect to the weights of the loss function is given by:

$$\boldsymbol{g} = \frac{1}{m}\nabla_{\boldsymbol{W}} \sum_{i=1}^{m} L(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)}, \boldsymbol{W}), \tag{A-7}$$

where ∇ is the vector differential operator. The gradients for the MLPs are calculated by using a backpropagation algorithm, where the loss is propagated backwards using the chain rule.

After calculating the gradient with respect to the weights, the weights are then updated by:

$$\boldsymbol{W} \longleftarrow \boldsymbol{W} - \epsilon \boldsymbol{g}, \tag{A-8}$$

where ϵ is the learning rate, which determines the rate at which the weights are updated. Eq. (A-7) and Eq. (A-8) are iterated until a minimum has been reached for the loss function in Eq. (A-4).

Calculating the gradients using the whole dataset can be computationally expensive when the dataset is large, this method is known as the batch gradient descent. To make the gradient descent more computationally feasible, the stochastic gradient descent can be used, where the real gradient of the batch gradient descent is approximated using a small set of the dataset. Using the small batch that is uniformly drawn from the whole dataset can reduce the computation time required for training.

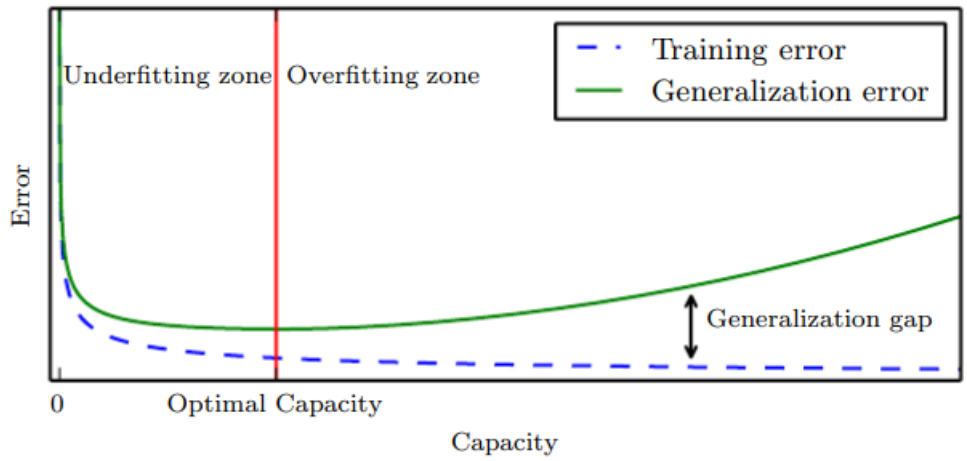


Figure A-3: Relation between generational and training error as a function of the model capacity. In the underfitting zone and starting with the lowest capacity, both generalization and training error will be high. And as the capacity of the model increases both error decreases until the optimal capacity has been reached. The overfitting zone starts when the generalization error starts increasing after hitting the plateau as the model capacity increases. Source:[11]

A-1-4 Capacity, Overfitting and Underfitting

For the deep learning model to perform well in the given task, the model has to perform well in predicting both the dataset which it was trained on and on unobserved datasets. Performing well on an unobserved dataset is called generalization. In order to measure the model performance on both types, the dataset is typically split into a training set and a test set which the model was not trained on. With the training set and a test set, the training error and the generalization error can be calculated. The generalization error is evaluated after the training is completed. To estimate the generalization error during the training, a validation set can be constructed from the training set. The validation set is not used for training and is evaluated after every training iteration.

From the training and generalization error, it can be determined if the trained model is underfitting or overfitting. Underfitting is when the model does not have a low enough training error and overfitting is when the error gap between the test and generalization error is large. Underfitting and overfitting is determined by the model's capacity. The model's capacity is the ability to fit a set of functions. With low capacity the model can overfit and with a high capacity the model can overfit. For a neural network, the capacity can be increased by adding more units and layers in the network, which allows the network to learn more complex functions. The general relationship between the generalization and training error as a function of the model capacity is shown in Figure A-3, the generalization error has a U-shape curve with the optimal capacity being at the center of the plateau. In order to obtain a good model with a low generalization error and perform well on unobservered inputs, it is required to find the appropriate model capacity for the complexity of the task.

Increasing or decreasing the model capacity of a deep learning by modifying the amount of units and layers in a network is not the only way of controlling the capacity of the model. In Section A-1-5 regularization techniques will be shown that can also control the model

capacity.

A-1-5 Regularization

Regularization is a collection of techniques to control the capacity of model, in Goodfellow *et al.*[11] regularization is defined as "Regularization is any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error", which is in a way of reducing the capacity and finding the optimal capacity. In this section two simple and popular regularization techniques will be shown: parameter norm penalty and early stopping.

Parameter norm penalty

Parameter norm penalty is a technique that reduces the capacity of the model by incorporating an extra cost penalty term on the weights in the loss function. The regularized loss function equation is given by:

$$J(\tilde{\boldsymbol{W}}) = J(\boldsymbol{W}) + \alpha\Omega(\boldsymbol{W}), \tag{A-9}$$

where $J(\boldsymbol{W})$ is the loss function, α is a chosen constant that controls the amount of penalty and $\Omega(\boldsymbol{W})$ is a penalty as a function of the model weights $\boldsymbol{W}$. Typically the penalty is only applied to the weights and not the bias constants. Depending on the choice of the parameter norm penalty $\Omega(\boldsymbol{W})$ will lead to a preference of a subset of functions from the set of functions which the unregularized model can fit, this has the effect of reducing the capacity of the model.

The most common penalty is the L^2 norm penalty which is known as weight decay and the function is given by:

$$\Omega(\boldsymbol{W}) = \frac{1}{2}||\boldsymbol{W}||_2^2 = \sqrt{\sum_i W_i^2}, \tag{A-10}$$

where W_i are the individual weight values.

Another penalty that can be used is the L^1 norm penalty, which is given by:

$$\Omega(\boldsymbol{W}) = ||\boldsymbol{W}||_1 = \sum_i |W_i|, \tag{A-11}$$

The regularization with both L^1 and L^2 will draw the optimal weights of the unregularized loss function closer to the origin, as seen on Figure A-4. The optimal weights for L^1 will be sparser than with L^2 regulizer, since the optimal weights can take a value of zero.

Early stopping

Early stopping is a method to stop during the training of the model to find a model with a low generalization error. When training a model with a capacity higher than the complexity of the task, the validation error will have U-shaped curve as a function of number of iterations, similar to the shape of the generalization error in Figure A-3.

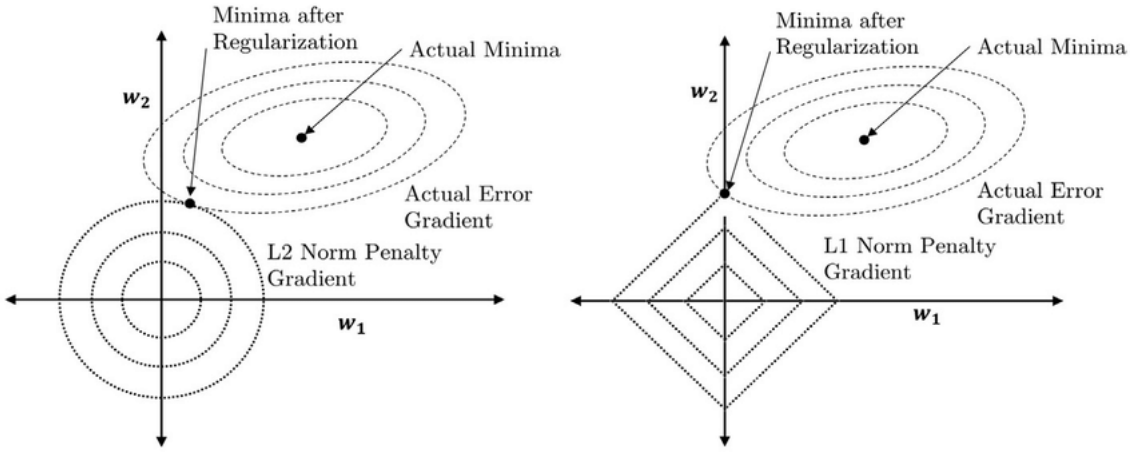


Figure A-4: Illustration of the effects of L^1 and L^2 regularization. The dotted elipses are the contours with equal value of the unregularized loss function and within it is its minima with its optimal weights. The dotted circles are the contours of equal for the the L^2 regularizer and dotted square contours for L^1 regularizer with their minima in the origin. When the regularization is applied to the loss function, a new minima will be formed between the contours of the unregularized function and the regularizer. Source: [12]

Early stopping works by saving the model weights at each iteration where the validation error is improved during the training and the training stops when the validation error error no longer improves over a set amount of iterations. The saved weights with the lowest validation error will then be used for the final model and hopefully a low generalization error. Early stopping is simple to implement and also has the benefit of reducing the computational cost of the training.

A-2 Extra figures

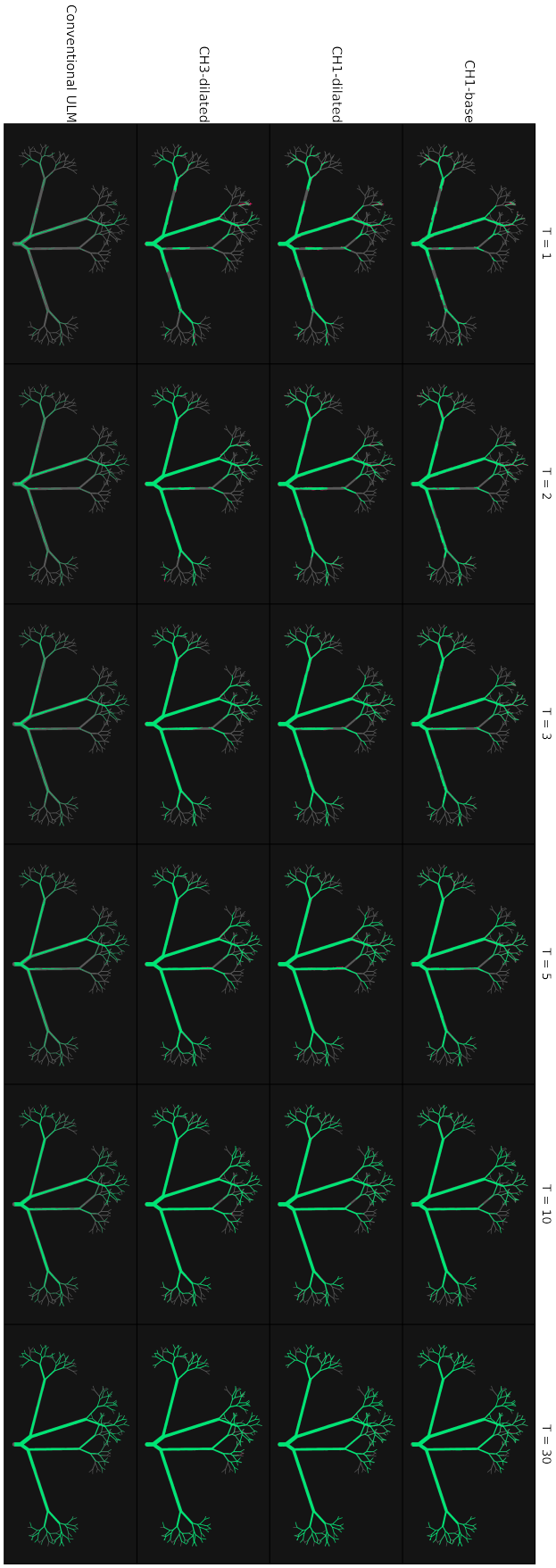


Figure A-5: Full Field of View (FOV) rat renal artery reconstruction at acquistion times T=1,2,3,4,5,30 using the trained models and conventional Ultrasound localization microscopy (ULM) method.

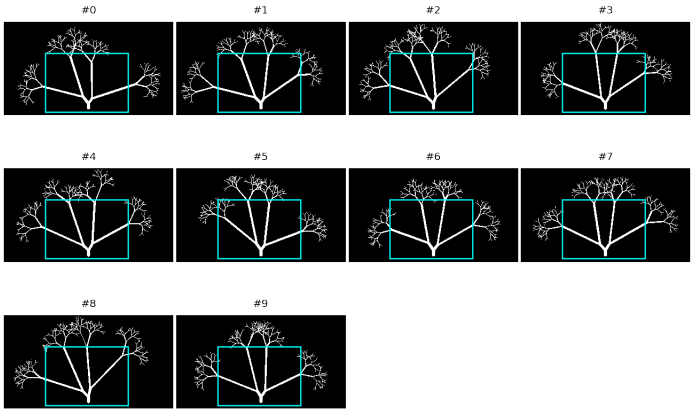


Figure A-6: Generated test samples with cyan region of interest (ROI) containing sparse large radii vessels and outside the ROI containing dense and small radii vessels.

Bibliography

[1] H. Peter, M. Kevin, and T. Abigail, *Diagnostic Ultrasound: Physics and Equipment*. Cambridge University Press, 2010.

[2] G. Montaldo, M. Tanter, J. Bercoff, N. Benech, and M. Fink, "Coherent plane-wave compounding for very high frame rate ultrasonography and transient elastography," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 56, no. 3, pp. 489–506, 2009.

[3] K. Christensen-Jeffries, O. Couture, P. A. Dayton, Y. C. Eldar, K. Hynynen, F. Kiessling, M. O'Reilly, G. F. Pinton, G. Schmitz, M.-X. Tang, *et al.*, "Super-resolution ultrasound imaging," *Ultrasound in medicine & biology*, vol. 46, no. 4, pp. 865–891, 2020.

[4] B. Heiles, A. Chavignon, V. Hingot, P. Lopez, E. Teston, and O. Couture, "Performance benchmarking of microbubble-localization algorithms for ultrasound localization microscopy," *Nature Biomedical Engineering*, vol. 6, no. 5, pp. 605–616, 2022.

[5] MathWorks, "What is a convolutional neural network?."

[6] J. Du, L. Wang, Y. Liu, Z. Zhou, Z. He, and Y. Jia, "Brain mri super-resolution using 3d dilated convolutional encoder–decoder network," *IEEE Access*, vol. 8, pp. 18938–18950, 2020.

[7] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: Lstm cells and network architectures," *Neural computation*, vol. 31, no. 7, pp. 1235–1270, 2019.

[8] A. Saguy, O. Alalouf, N. Opatovski, S. Jang, M. Heilemann, and Y. Shechtman, "Dblink: Dynamic localization microscopy in super spatiotemporal resolution via deep learning," *Nature Methods*, pp. 1–10, 2023.

[9] D. A. Nordsletten, S. Blackett, M. D. Bentley, E. L. Ritman, and N. P. Smith, "Structural morphology of renal vasculature," *American Journal of Physiology-Heart and Circulatory Physiology*, vol. 291, no. 1, pp. H296–H309, 2006.

[10] M. Banoula, "What is perceptron: A beginners guide for perceptron?."

[11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[12] K. Santosh, N. Das, and S. Ghosh, *Deep learning models for medical imaging*. Academic Press, 2021.

[13] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation functions: Comparison of trends in practice and research for deep learning," *arXiv preprint arXiv:1811.03378*, 2018.

[14] O. Couture, V. Hingot, B. Heiles, P. Muleki-Seya, and M. Tanter, "Ultrasound localization microscopy and super-resolution: A state of the art," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 65, no. 8, pp. 1304–1320, 2018.

[15] R. J. van Sloun, O. Solomon, M. Bruce, Z. Z. Khaing, H. Wijkstra, Y. C. Eldar, and M. Mischi, "Super-resolution ultrasound localization microscopy through deep learning," *IEEE transactions on medical imaging*, vol. 40, no. 3, pp. 829–839, 2020.

[16] L. Milecki, J. Porée, H. Belgharbi, C. Bourquin, R. Damseh, P. Delafontaine-Martel, F. Lesage, M. Gasse, and J. Provost, "A deep learning framework for spatiotemporal ultrasound localization microscopy," *IEEE Transactions on Medical Imaging*, vol. 40, no. 5, pp. 1428–1437, 2021.

[17] X. Chen, M. R. Lowerison, Z. Dong, N. V. C. Sekaran, D. A. Llano, and P. Song, "Localization free super-resolution microbubble velocimetry using a long short-term memory neural network," *IEEE Transactions on Medical Imaging*, 2023.

[18] M. Maqbool, *An introduction to medical physics*. Springer, 2017.

[19] H. Azhari, *Basics of biomedical ultrasound for engineers*. John Wiley & Sons, 2010.

[20] R. S. Cobbold, *Foundations of biomedical ultrasound*. Oxford university press, 2006.

[21] O. Couture, B. Besson, G. Montaldo, M. Fink, and M. Tanter, "Microbubble ultrasound super-localization imaging (musli)," in *2011 IEEE International Ultrasonics Symposium*, pp. 1285–1287, IEEE, 2011.

[22] O. Viessmann, R. Eckersley, K. Christensen-Jeffries, M.-X. Tang, and C. Dunsby, "Acoustic super-resolution with ultrasound and microbubbles," *Physics in Medicine & Biology*, vol. 58, no. 18, p. 6447, 2013.

[23] K. Christensen-Jeffries, S. Harput, J. Brown, P. N. Wells, P. Aljabar, C. Dunsby, M.-X. Tang, and R. J. Eckersley, "Microbubble axial localization errors in ultrasound super-resolution imaging," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 64, no. 11, pp. 1644–1654, 2017.

[24] D. Ackermann and G. Schmitz, "Detection and tracking of multiple microbubbles in ultrasound b-mode images," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 63, no. 1, pp. 72–82, 2015.

[25] C. Errico, J. Pierre, S. Pezet, Y. Desailly, Z. Lenkei, O. Couture, and M. Tanter, "Ultrafast ultrasound localization microscopy for deep super-resolution vascular imaging," *Nature*, vol. 527, no. 7579, pp. 499–502, 2015.

[26] K. Christensen-Jeffries, R. J. Browning, M.-X. Tang, C. Dunsby, and R. J. Eckersley, "In vivo acoustic super-resolution and super-resolved velocity mapping using microbubbles," *IEEE transactions on medical imaging*, vol. 34, no. 2, pp. 433–440, 2014.

[27] Y. Desailly, J. Pierre, O. Couture, and M. Tanter, "Resolution limits of ultrafast ultrasound localization microscopy," *Physics in medicine & biology*, vol. 60, no. 22, p. 8723, 2015.

[28] V. Hingot, C. Errico, B. Heiles, L. Rahal, M. Tanter, and O. Couture, "Microvascular flow dictates the compromise between spatial resolution and acquisition time in ultrasound localization microscopy," *Scientific reports*, vol. 9, no. 1, p. 2456, 2019.

[29] Q. Chen, H. Song, J. Yu, and K. Kim, "Current development and applications of super-resolution ultrasound imaging," *Sensors*, vol. 21, no. 7, p. 2417, 2021.

[30] S. Harput, K. Christensen-Jeffries, J. Brown, Y. Li, K. J. Williams, A. H. Davies, R. J. Eckersley, C. Dunsby, and M.-X. Tang, "Two-stage motion correction for super-resolution ultrasound imaging in human lower limb," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 65, no. 5, pp. 803–814, 2018.

[31] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, *et al.*, "Recent advances in convolutional neural networks," *Pattern recognition*, vol. 77, pp. 354–377, 2018.

[32] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.

[33] D. Müller, I. Soto-Rey, and F. Kramer, "Towards a guideline for evaluation metrics in medical image segmentation," *BMC Research Notes*, vol. 15, no. 1, p. 210, 2022.

[34] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2623–2631, 2019.

[35] N. Blanken, B. Heiles, A. Kuliesh, M. Versluis, K. Jain, D. Maresca, and G. Lajoinie, "Proteus: A physically realistic contrast-enhanced ultrasound simulator—part i: Numerical methods," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 72, no. 7, pp. 848–865, 2024.

Glossary

List of Acronyms

RF	Radio Frequency
SNR	Signal to Noise ratio
ULM	Ultrasound localization microscopy
PSF	Point spread function
MB	microbubble
FWHM	full-width at half maximum
MLPs	multilayer perceptrons
CNN	Convolutional Neural Networks
RNN	Recurrent Neural Networks
LSTM	long short-term memory
SMLM	single-molecule localization microscopy
s.d.	standard deviation
RFd	receptive field
MSE	mean squared error
TV	total variance
BCE	binary cross-entropy
TPE	Tree-structured Parzen Estimator
FOV	Field of View
ROI	region of interest

List of Symbols

∇	Vector differential operator

λ	Wavelength (in m)
ρ	Medium density (in kg/m^3)
c	Sound of speed (in m/s)
f	Frequency (in Hz)
P	Pressure (in Pascal)
P_0	2-3
R_L	The minimum distance which two separate points can still be distinguished (in m)
u_p	Particle velocity (in m/s)
Z	Acoustic impedance (in Rayls or kg/m^2/s)