Generation of Synthetic MRI with Deep Optical Flow Field Estimation for Faster Imaging

by
Hosein Neeli

A Thesis submitted to the Department of Computer Science,

College of Natural Sciences and Mathematics

in partial fulfillment of the requirements for the degree of

Master of Science
in Computer Science

Chair of Committee: Dr. Nikolaos V. Tsekos, University of Houston

Committee Member: Dr. Giulia Toti, University of Houston

Committee Member: Dr. Alexander Mamonov, University of Houston

University of Houston
December 2019

"Yesterday I was clever, so I wanted to change the world.

Today I am wise, so I am changing myself".

-Rumi

ACKNOWLEDGMENTS

My thesis would not have been possible to accomplish without the guidance and help of my advisor, Prof. Nikolaos V. Tsekos. I would like to express my gratitude and appreciation to him for being an awesome advisor. He taught me how to look at problems in different aspects and how to dynamically think about the solutions. I learned from Prof. Tsekos to be confident about myself and push my limits.

I cannot express enough thanks to my committee: Dr. Giulia Toti and Dr. Alexander Mamonov for their valuable comment. Their experience and technical views significantly improved my thesis.

I would like to thank my colleagues in the Medical Robotics and Imaging Laboratory at the University of Houston, Daniel Velazco, Giovanni Molina, Adina Micula, Cristina Morales, and Wenhui Chu, for their help and support. They made a fun and friendly environment that was my motivation to study and research.

I would like to take this opportunity to express my gratitude to my friends in Houston, Hesam Moradi, Milad Heydariaan, Amir Hossein Rahmati, Millad Ghane, Masoud Poshtiban, Vida Tabrizi, Pushpendra, and Sorena Sarmadi, and in Tehran, Raza Nazari, for their unconditional help.

I like to especially thank my aunt, Mrs. Dodd for letting me stay in her home during my master program.

Finally, my deepest gratitude to my mom, Zohreh, my dad, Javad, my brother, Vahid, and my lovely girlfriend, Farzaneh. They are always my strongest supporters from 7,500 miles away in Tehran. At this time that I am writing this acknowledgment, the internet has been completely shut down in Iran, and I am not able to tell them how much I love them.

ABSTRACT

Magnetic Resonance Imaging (MRI) is an effective, non-invasive, and revolutionary imaging technique used to diagnose, study, and analyze chemical and physical structures inside the body. MR image acquisition suffers from two significant problems. First of all, prolonged scanning sessions are inconvenient and costly for patients. Secondly, the respiratory motion of the patients or external noise cause artifacts. Addressing these two issues is a strong motivation for making MRI procedures faster and more accurate. A wide variety of methods have been used to shorten the MR image acquisition process by optimizing current techniques or improving the mechanical and computational performance of the scanners.

An appealing solution for the mentioned problems is to scan fewer MR images and generate in-between images to make the MR image acquisition faster. Also, in the presence of motion artifacts, we can reconstruct the imperfect images, if such a technique is available.

In this thesis, we trained and applied a deep learning model to synthesize an arbitrary number of intermediate MR images by estimating optical flow vectors of two consecutive MRI slices or frames. We investigated the performance of synthetic MR images produced by this method and compared them with one of the available-to-use related methods. The evaluation results show that this technique produces high quality multiple intermediate images and outperforms the related method.

TABLE OF CONTENTS

1 Introduction **1**

2 Background **4**
2.1 Magnetic Resonance Imaging 4
2.2 Artificial Intelligence 4
2.3 Applications of Artificial Intelligence in Medical Science 5
2.4 Optical Flow 6

3 Related Work **8**
3.1 Faster MRI 8
3.2 Image Interpolation 8
3.3 Optical Flow Estimation 9
3.4 Image Reconstruction 9

4 Design **10**
4.1 Arbitrary-Time Optical Flow Estimation 12
4.2 Network Details 12
4.3 Data set 13
4.3.1 Data Augmentation 15
4.3.2 Refining Data set 16
4.4 Training our Neural Network 17
4.5 Loss Functions 18
4.6 Training Tools and Resources 20

5 Evaluation **22**
5.1 Evaluation Methods 22
5.1.1 Peak Signal-to-Noise Ratio 22
5.1.2 Mean Squared Error 23
5.1.3 Structural Similarity Index 23
5.1.4 Pixel by Pixel Difference 24
5.2 Evaluation Scenarios 24
5.2.1 Temporal Single-Frame Synthesis 26
5.2.2 Spatial Single-Slice Synthesis 29
5.2.3 Temporal Multi-Frame Synthesis 31
5.2.4 Spatial Multi-Slice Synthesis 34
5.2.5 Coherency Challenge 38
5.2.6 Frame Synthesis Based on Synthesized Frames 39
5.2.7 CINE MRI Video Frame Synthesis 41

6 Conclusion **42**

Bibliography **43**

Appendix **48**

LIST OF TABLES

1 Training data set configuration. 14
2 The results of temporal single-frame synthesis. 28
3 The results of spatial single-slice synthesis. 31
4 The results of temporal multi-frame synthesis. 33
5 The results of spatial multi-slice synthesis. 37
6 The results of slice interpolation based on synthetic slices. 40

LIST OF FIGURES

1 Respiratory cardiac motion artifacts in abdominal MRI. 2
2 Synthesizing arbitrary in-between magnetic resonance frames or slices. 3
3 Optical flow vectors in cardiac imaging. 7
4 U-Net network configuration. 14
5 Flow diagram of the end-to-end network. 18
6 Spatial slice spacing. 25
7 Temporal difference. 25
8 Temporal single-frame synthesis evaluation scenario. 26
9 Temporal single-frame comparison results. 27
10 Temporal single-frame visual comparison. 28
11 Spatial single-slice synthesis evaluation scenario. 29
12 Spatial single-slice comparison results. 30
13 Spatial single-slice visual comparison. 30
14 Temporal multi-frame synthesis evaluation scenario. 32
15 Temporal multi-frame comparison results. 32
16 Temporal multi-frame visual comparison. 34
17 Spatial multi-slice synthesis evaluation scenario. 35
18 Spatial multi-slice comparison results. 36
19 Spatial multi-slice visual comparison. 37
20 Coherency problem. 39
21 Evaluation of slice synthesis based on synthetic slices. 40
22 Visual comparison of slice synthesis based on synthetic slices. 41
23 Comparison of temporal single frame generation. 48
24 Comparison of Spatial single slice generation. 49
25 Comparison of temporal multi frame generation. 50
26 Comparison of Spatial multi slice generation. 51

1 Introduction

Performance and image quality of Magnetic Resonance Imaging (MRI) have been improved significantly in recent decades. High-quality MRI helped physicians to increase their ability to study structural and chemical structure of the body and diagnose diseases. However, two crucial problems have remained unsolved. First of all, MR image acquisition is a prolonged process that is inconvenient for patients. The process can take more than an hour to be completed. During this process, the patient must stay in the narrow and tight donut-shaped magnet tube. Also, the rotation of the sensors around the central axis of the scanner makes a frightening noise that is insufferable for sensitive patients such as children and older adults.

The other problem is the motion artifacts that are created by movement or breathing of the patients. During the image acquisition, the radiologist asks patients to hold their breath to avoid the respiratory motion of the body. Capturing images during body motion could make blurry and imperfect images, which are not reliable for medical interpretation (see Figure 1). The images are not always perfect even when patients are asked to hold their breath and stay still in the tube. The reason is that some patients are not able to hold their breath for a long time, or scary noise of rotating scanner may make them have unintentional motions inside the tube. Even a heartbeat creates motion artifact on MR images. In this case, they are asked to re-enter the tube and redo the scanning process again, which is very costly and frustrating for the patients.

It is crucial to address the mentioned problems because, for the first problem, by faster MR image acquisition, patients can be discharged from the MRI scanner tube quicker with less inconvenience. Moreover, medical imaging clinics can financially benefit by accommodating more patients in a time slot, and perhaps they can decrease MRI costs per patient. Furthermore, by solving the motion artifact problem, there is no need to keep patients in the scanner tube to be re-scanned.

There are multiple proposed techniques for faster MRI that improve performance by increasing the number of coils or employing multiple arrays of sensors. However, these methods are tough to develop because of costly research and development.

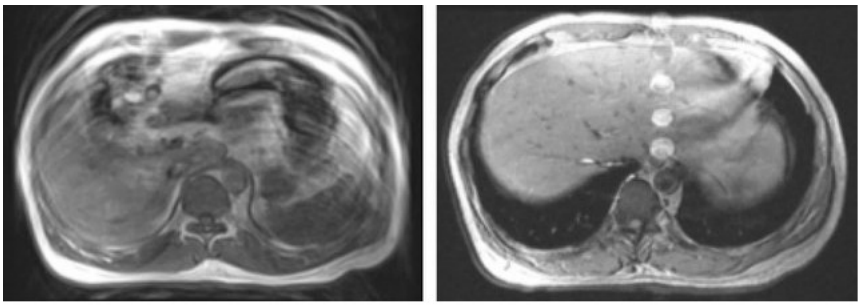

Figure 1: Respiratory cardiac motion artifacts in abdominal MRI. (a) high motion artifacts caused by respiration in an MRI session without breath-holding. (b) represents minimal respiratory motion artifact in a breath-holding MRI session (check Zhuo and Gullapalli [43]).

Most current practices for faster MRI by scanning fewer images focus on single image synthesis. These methods are slow and not able to synthesize multiple intermediate slices at a time because they have to use a synthetic slice recursively to synthesize more than one slice. Also, because every synthetic slice has a certain amount of error, the recursive method propagates the error for multiple slice synthesis.

It is very challenging to synthesize intermediate MR images when slice spacing and temporal difference (Figures 6 and 7) is high because slices and frames are not spatially and temporally coherent. Moreover, a proper solution must address the occlusion problem where a part of the body has motion; otherwise, it could produce artifacts near motion boundaries.

In our research, we optimized a video interpolation method ([18]) for the generation of an arbitrary number of intermediate MR images (Figure 2). The method is based on estimation of motion vectors of the pixels in two consecutive MR input images by optical flow technique and using calculated motion (flow) vectors to translate pixels in any arbitrary time step between input images. We addressed the problem of disappearing pixels between two images (occlusion problem) by introducing visibility maps (Equation 2). We use a convolutional neural network (CNN) to estimate the motion vectors of the two base images. Then we use another CNN to enhance the quality of the interpolated image in motion boundaries by calculating visibility maps.

In this thesis, we make the following contributions:

- We design and optimize a video multi-frame synthesis method based on optical flow for the first time for Magnetic Resonance Image synthesis. We call it Fast-MRI.
- We evaluate the performance of Fast-MRI in multiple scenarios and compare it with one of the accessible state-of-the-art related methods.

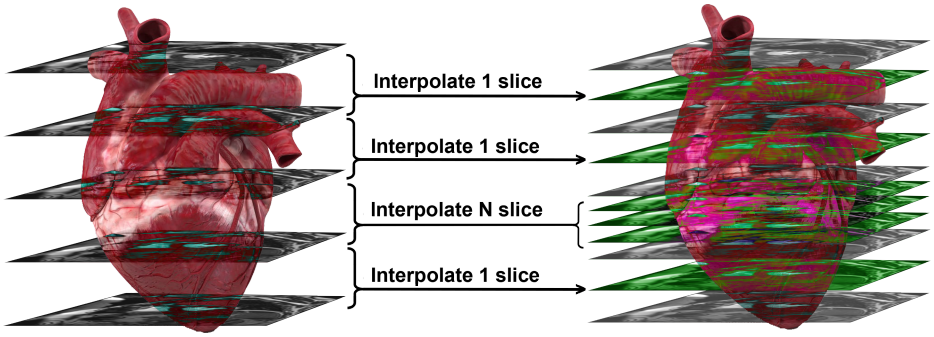

Figure 2: Synthesizing arbitrary in-between magnetic resonance frames or slices.

2 Background

2.1 Magnetic Resonance Imaging

The magnetic resonance phenomenon was discovered by Purcell and Block independently in 1946. In the early 70s, Damadian used the phenomenon to invent an MRI scanner. MRI is a non-invasive and painless method for the acquisition of high-quality images from body tissues. Huge magnets magnetize the protons by a strong magnetic field, and multiple radio frequency sensors accurately measure the amount of energy released by magnetized protons. The captured data is sent to a computer, and then the computer performs a large amount of mathematical computation to convert the received data to a human-understandable image. Scientists use MR images to study and diagnose various kinds of diseases [40].

2.2 Artificial Intelligence

Have you ever been impressed by how Google search engine sorts the search results related to your interests? Or how close the products suggested by Amazon's online shopping website are to your daily needs? Nowadays, these products and information are not randomly suggested. A smart decision-making mechanism lies behind the system to learn your online behavior and fit the products, services, and whatever they try to present to your needs and interests.

It is good to mention other use cases of artificial intelligence in our life to make a better understanding of the subject. As a hot example, autonomous driving automobiles are built entirely based on AI. They use advanced Computer Vision techniques to scan and sense the condition of roads. Based on the perceived data and what they have learned from real human driving abilities, they can decide when to accelerate, brake, or turn. These automobiles distinguish stationary objects from pedestrians and are able to stop immediately to avoid collisions. Autonomous cars are even able to understand road signs and take proper actions.

With artificial intelligence and its subcategories, i.e., machine learning and deep learning, we

teach computers to think like the human brain. We try to give the ability to computers to automatically classify, decide, or act as we do in our everyday life. Teaching human-like capabilities to computers needs a vast number of examples, a.k.a. labeled data. Computers learn the decision or reaction of humans to events by looking at the given examples. Multiple subcategories of artificial intelligence have emerged in recent years. Machine learning and deep learning are the most prominent examples.

In machine learning, we manually introduce essential features of data to the computers and make it classify or decide base on the learned features. However, in deep learning, we design a machine to automatically extract features from the given data and connect these features to their corresponding labels. Like a small baby, the machines must observe a large number of examples to gradually learn.

2.3 Applications of Artificial Intelligence in Medical Science

From the day that computer-aided diagnosis/detection (CAD) were invented in the 60s, biologists and physicians have been trying to collaborate with computer scientists to employ computers to study, diagnose, and cure diseases and disorders. This process became serious when computers got the ability to scan, process, and analyze images. Biologists had been using visual techniques to diagnose abnormalities and disease in analog medical scanned images for a long time before the emergence of computer-aided image processing. There were many misdiagnoses in their work because the Human Visual System (HVS) is not powerful enough to distinguish tiny details or particles in low-quality medical images or ones that carry noise or motion artifacts. Medical imaging analysis systems were developed from the late 70s to the mid-90s. They designed simple pixel-level line, edge, and shape detection models. By the current scale of knowledge, we cannot call them automatic analytical systems anymore, because they utilized a series of consecutive conditional statements to process pixels in medical images. The correct terminology to call them is rule based systems instead of smart systems [24]. In the late 90s, supervised computer models were developed for image segmentation in order to be used in medical diagnosis purposes. As an example, those

models were able to determine decision boundaries in high-dimensional environments [24]. Although these methods are widely used nowadays, at that time, the lack of computer processing power prevented them from being popular. Mathematical scientists started working on models by which computers could automatically learn features of the medical images. This is the infrastructure of deep learning models that are popular in the current era. One of the prominent efforts on this concept was done by Fukushima [9] in 1980. He introduced a convolutional neural network for pattern recognition. Moreover, in 1995, Lo et al. [26] applied a similar method for lung nodule detection. Nowadays, artificial intelligence, particularly deep learning is widely used in every aspect of medical science, i.e., image acquisition, image enhancement and reconstruction, diagnosis and even individual treatment design.

2.4 Optical Flow

In a single image, spatial information of objects is captured. However, by having a sequence of images from a scene, we can understand the temporal information of the objects. By optical flow, we can track the apparent motion of the objects relative to a camera or an observer in a sequence of images [13]. Optical flow finds the displacement vector of a pixel or a group of pixels (an object) in a sequence of motion pictures or videos. By having the optical flow vector (a.k.a. motion vector or flow vector) for a pixel in two consecutive frames, we can map the location of a pixel from the first frame to its location in the second one. Figure 3 visualizes the motion vectors of the wall of the left ventricles in two consecutive frames of a cardiac CINE MRI (see Section 5.2.7).

Optical flow has several applications in various fields of science. Object tracking is the most important application. Video compression and stabilization are among the growing techniques implemented by optical flow.

We should have two following main assumptions to calculate optical flow [1, 13].

1. Pixel intensities do not immediately change between consecutive frames.

2. Groups of pixels move together. The optical flow works very well where the motion is smooth,

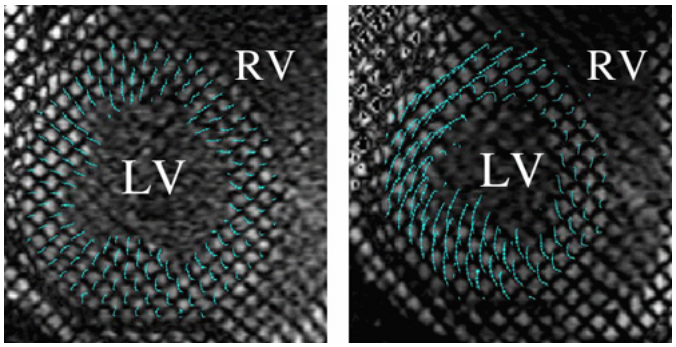

Figure 3: Optical flow vectors in cardiac imaging. Displacement vectors of the wall of the left ventricle of the heart (in blue) shows the direction and magnitude of the motion [41].

and the pixels do not jump from a place to another in consecutive frames.

3 Related Work

3.1 Faster MRI

In one of the most important efforts to reduce the MRI scanning time, Lusting et al. [27] proposed a compressed sensing MRI, which is known as HyperSense. This state-of-the-art technique reduces MR image acquisition time by 30-50% according to an online article by GE Healthcare [32]. HyperSense makes MRI faster by scanning less data and applying a technique based on compressed sensing to reconstruct missing data without significant quality loss.

Feinberg and Kawin [27] proposed a technique for multi-slice brain MR image acquisition. Pruessmann et al. [34] proposed a method to improve sensitivity encoding for faster MRI.

3.2 Image Interpolation

Image interpolation, especially video frame interpolation by optical flow, has been a hot topic for a long time. The performance of approaches proposed by Herbst et al. [12] and Barron et al. [2] by looking at the interpolation error values are impressive, but their methods suffer from an occlusion problem. Occlusion happens when a pixel or an object frame is present in one of the two consecutive images and disappears in the other one. The occlusion problem results in artifacts in motion boundaries in an image. For video interpolation, especially in high frame rate videos, a high amount of artifact does not make a significant visual impact. However, in MR imaging, mostly in cardiac MRI, it is crucial to accurately recreate moving edges because those locations indicate defects and failure in the heart. Mahajan et al. [28] utilize image gradients to synthesize intermediate images. This method is computationally costly due to the complex technique that they use for optimization. Meyer et al. [29] use a phase information processing technique in a multi-level pyramid of the images, but it can poorly interpolate between images with large movements.

3.3 Optical Flow Estimation

In recent years, optical flow estimation methods based on convolutional neural network has become very popular. As an example, Dosovitsky et al. [8] proposed two learning models for mapping consecutive video frames to the respective optical flow matrix employing a deep learning method, and Ilg et al. [16] improved the performance of this deep learning method by optimizing the model.

We categorize the methods mentioned above as supervised machine learning techniques. Jason et al. [17] designed an unsupervised model to estimate arbitrary-time optical flow vectors in order to use in frame warping.

Liang et al. [22] designed another optical flow-based technique, but this time for extrapolation of a frame of a video sequence. The authors used estimated flow by another network [35] for their training process.

Niklaus et al. [30] introduced a video frame interpolation by using a convolutional neural network for local convolution over two input images. This CNN learns spatially adaptive kernels for each pixel. This method produces high-quality frames. However, it is not computationally efficient and needs enormous amounts of memory for the estimation of convolutional kernels for each pixel. Later they enhanced the memory efficiency of their method by separable kernels (SepConv) [31].

3.4 Image Reconstruction

Researchers proposed a wide variety of methods to reconstruct or synthesize MR images. Samadi-Miyandoab et al. [37] proposed a method of image interpolation based on an optical flow of CT scan images for five real lung cancer patients. They used DIRART software to calculate vector fields for optical flow calculation. Moreover, Lin et al. [23] introduced a decomposition-reconstruction interpolation method based on artificial neural network to reconstruct defected or missing MRI slices. The current work is categorized in this category as it tried to propose a reconstruction method through a deep learning technique.

4 Design

Every pixel in two consecutive frames has a direction and magnitude of movement. We can estimate the location of the pixels at any time between two successive frames by translation of the original locations by its corresponding motion vector to their locations in the next frame. In the papers in the field of optical flow computation, the mentioned translation of the pixels is called warping. We calculate linear motion vectors between consecutive frames by optical flow [13].

If we want to know the location of pixels in any frame in-between two consecutive frames, we should calculate the motion vectors from pixels in any of the frames to the intermediate frame. But the intermediate frame is not available for calculation of the intermediate frame. As a simple solution for this problem, we can compute the motion vectors between the consecutive frames and divide it proportional to the distance of the intermediate frame to each one of the captured frames (bi-directional motion vectors) and warp the intermediate frame (see Equation 1). Warped intermediate frames by this approach are not accurate. The reason is that we divide a linear motion vector for warping, but in nature, objects do not have linear motion. We train a deep convolutional neural network to accurately estimate bi-directional flow vectors.

Jiang et al. [18] proposed a method for video frame interpolation using optical flow computation (Equation 2). This method uses deep convolutional neural networks for learning various features of a sequence of video frames such as bi-directional optical flow and visibility maps to interpolate arbitrary number of intermediate frames. We optimized this method for MR image interpolation and assessed its performance in the synthesis of temporal and spatial MR images. This is the first time that this method has been used on MR images. We call this method Fast-MRI.

- We completely redesigned and implemented the data pre-processing part of the original model to prepare and process MRI data.
- We changed the hyper-parameters, and the configuration of the CNN in order to optimize the model and learn optical flow features of MR images.

- We prepared a massive MRI data set for the network training.

Assume we have two base frames I_0 and I_1, our goal is to synthesize, or technically speaking, interpolate, multiple intermediate frames ($\hat{I}_t$) in an arbitrary time between $t = 0$ and $t = 1$, based on two given frames also known as base frames.

Jiang et al. [18] introduced an intermediate video frame synthesis method that achieves the mentioned goal by estimating intermediate frames by bi-directional optical flow vectors and combining them. This method considers multiple parameters to enhance synthesis, such as disappearing objects and smoothness of motion boundaries in a video sequence.

In the below description of the method, $F_{x\to y}$ denotes the optical flow from $I_{t=x}$ to $I_{t=y}$. If we have I_0, I_1, $F_{t\to 0}$, and $F_{t\to 1}$, we can interpolate the intermediate frame $\hat{I}_t$ for any t that $0 < t < 1$ by the following Equation.

$$\hat{I}_t = \alpha_0 \odot g(I_0, F_{t\to 0}) + (1 - \alpha_0) \odot g(I_1, F_{t\to 1}) \tag{1}$$

where g is warping function that maps the pixels in I_0 by the calculated optical flow vectors ($F_{t\to 1}$) to their location in I_1. α_0 is the weighting mechanism that indicated the contribution of the base frames based on the ongoing arbitrary t^{th} frame between I_0 and I_1. As an example, if the selected t is closer to 1, the contribution of I_1 is larger to the final interpolated image. $\odot$ denotes element-wise multiplication and is used for weighting of the base images to the synthetic image [18].

In video frame synthesis, a challenging problem happens when an object is present in the first frame then disappears in the next one; this problem is called occlusion. This phenomenon usually happens around motion boundaries. To tackle this problem, Jiang et al. [18] introduced visibility maps which are masks for every pixel of base images. $V_{t\leftarrow x}$ is 0 when the pixel is not visible when we move from time x to time t and 1 when it is still visible. We re-write the Equation 1 by adding the visibility maps.

$$\hat{I}_t = \frac{1}{M} \cdot ((1 - t)V_{t\leftarrow 0} \cdot g(I_0, F_{t\to 0}) + tV_{t\leftarrow 1} \cdot g(I_1, F_{t\to 1})) \tag{2}$$

where $M = (1-t)V_{t\leftarrow 0} + tV_{t\leftarrow 1}$ is defined as a term of normalization.

4.1 Arbitrary-Time Optical Flow Estimation

In Equation 2, we use optical flow between intermediate frames and base frames to estimate the $\hat{I}_t$, but in reality, we have no access to this optical flow value $F_{t\to 0}$ and $F_{t\to 1}$. To solve this challenge, Jiang et al. [18] estimate $F_{t\to 0}$ and $F_{t\to 1}$ using $F_{0\to 1}$ and $F_{1\to 0}$ by simply dividing the flow vectors proportional to the arbitrary time t $(0 < t < 1)$. The formulae for arbitrary-time optical flow calculation are presented below:

$$F_{0\to t} = t.F_{0\to 1} \tag{3}$$

$$F_{1\to t} = -(1-t).F_{1\to 0} \tag{4}$$

4.2 Network Details

The estimation of optical flow vectors at an arbitrary time in-between two frames (Equations 3 and 4) is expected to have good results in smooth regions but not in motion boundaries. To address this limitation, by the first CNN, we estimate linear motion vectors ($F_{0\to 1}$ and $F_{1\to 0}$), then compute bi-directional optical flows, and by the second CNN we refine estimated flow vectors by estimation of residuals that we should add to $F_{t\to 0}$ and $F_{t\to 1}$.

As we discussed before, visibility maps are essential to handle occlusion problem, so we predict two visibility maps $V_{t\leftarrow 0}$ and $V_{t\leftarrow 1}$ using the second learning part of the interpolation network.

To make the second part of the interpolation network up and running, we need to have bi-directional optical flow matrices. So in the first part of the network, we use base frames I_0 and I_1 to estimate $F_{0\to 1}$ and $F_{1\to 0}$. The U-Net model presented by Ronneberger et al. [36] is used to form a bi-directional optical flow estimation network. The U-Net is a fully convolutional neural network in which an encoder and a decoder are used. It uses several skip connections between the layers of encoder and decoder with equal spatial resolution to avoid vanishing gradients. As

illustrated in Figure 4, U-Net [36] has six hierarchies in the encoder and five in the decoder. Each of the hierarchies in the encoder is created by two consecutive convolutional layers following a Leaky ReLU with the alpha number of 0.1. Then comes an average pooling layer ($stride = 2$) for the decrease of spatial dimension. In the decoder part, U-Net uses bi-linear up-sampling by factor 2 following two convolutions and a Leaky ReLU activation function for all five hierarchies.

For the bi-directional optical flow computation, which is the first part of the learning network, Jiang et al. [18] changed the default kernel size of U-Net. They set 7×7 kernels in the first hierarchy of convolutional layers following 5×5 kernels in the second one and 3×3 for the rest; however, for learning more features from low resolution MR images, we set all the kernel sizes to 3×3 and the stride number to 1 for the convolutions. The stride number for pooling remained 1. As illustrated in Figure 5, the computation part calculates arbitrary-time optical flows based on Equations 3 and 4. Then we create two warped images using the outputs of previous step and the respective base frames I_0 and I_1. The second network requires warped images as input. From this computation step, we formed the matrices to feed the arbitrary-time optical flow synthesis network. The output of the second part of the network pipeline is visibility maps and arbitrary-time optical flow matrices. We had the base frames and every term of interpolation Equation 2, so we create the interpolated frame $\hat{I}_t$.

4.3 Data set

One of the most critical issues that biologists and computer scientists are dealing with is the lack of data sets for their research in medical image processing. Naturally, medical images are hard to obtain, and using them has strict restrictions.

- Using MR images requires the consent of the patients and medical imaging institutes. Obtaining such permissions is usually very hard and has a long process.

- To keep patients anonymous, some identifying parts of the subjects are masked, which makes the data set useless for some applications like unsupervised learning because the masks are

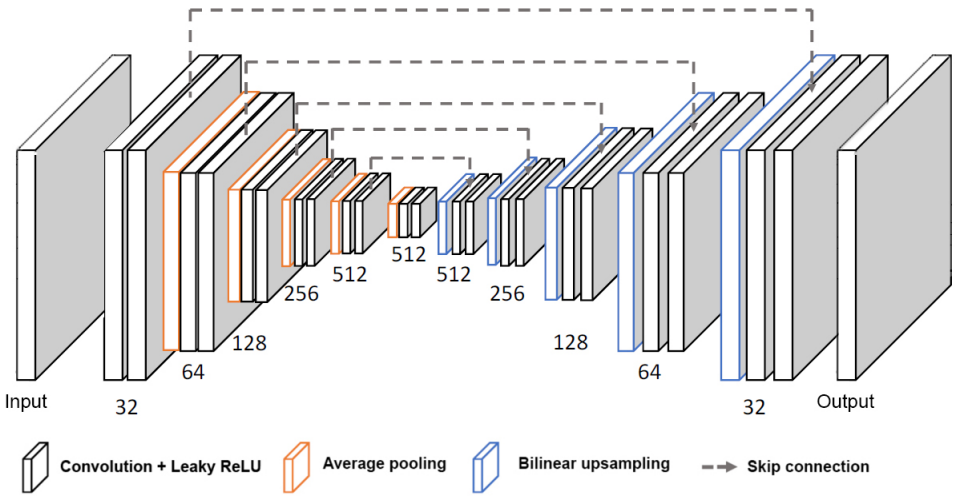

Figure 4: U-Net network configuration. U-Net is used as the main architecture for the first and second part of the learning network; it consists of six hierarchies of convolutional layers in the encoder and five in the decoder part.

not coherent.

For training our convolutional neural network, we used the Adobe 240-fps high frame rate videos data set [38] containing 79,700 frames alongside the YouTube 240-fps [18] that contains 296,300 consecutive frames. Both data sets contain 240 frames per second videos [18]. Detailed information about these data sets can be found in Table 1.

Table 1: Training data set configuration.

data set name	**Adobe 240-fps**	**YouTube 240-fps**	**The Sunnybrook**
Number of frames	79,768	185,352	25,207
Resolution (pixel)	1280×720	1280×720	Max 256×256
Frames per second (fps)	240	240	Max 80
Is it medical data?	No	No	Yes
About 8,000 un-usable images were dropped from the Sunnybrook data set.			

The optical flow is more precisely calculated in regions with small motions [1, 13]. Each MRI spatial slice or temporal frame, even with new scanning and reconstructing methods take optimistically about 60 milliseconds so, the MRI scanner can scan up to 16 frames in a second [4].

On the other hand, 240 fps videos contain motion information of a scene in every 4.1 milliseconds. Compared to every 60 milliseconds in MRI images, the motions in a video sequence are smaller. Thus, we can extract more optical flow information from the small motions in 240 fps videos to train the model. You can find the main reason for using a non-MRI data set for training our model.

- A 240-fps video data set captures smaller motions that yield more precise optical flow fields [1].
- Deformation in human body tissues due to respiration, heartbeat, movement, and other human body activities during 60 milliseconds is drastically higher than the Adobe 240-fps data set.
- The probability of having unwanted motion artifacts is higher when the time of image acquisition for a single slice/frame is high. Motion artifacts cause inaccurate training data and considered outliers.

One of the reasons that MRI slices/frames are slow to scan is that the procedure of the MR image acquisition is not like regular photography, which is capturing color intensities produced by the reflection of light on a CMOS sensor. The MRI scanner captures the changes in the orientation of particles in body tissue affected by a strong magnetic field [3]. The captured data is different from color intensity. The data is captured as a huge matrix of complex numbers in the frequency space a.k.a. K-Space. This data must be pre-processed, be converted and reconstructed as an understandable format for the human visual system. This process needs high computational power, which takes substantial time in the scale of medical imaging. New advancements in mathematical methods and computational power improvements decreased the reconstruction time for each slice/frame from 240 milliseconds to 60 milliseconds for an average 256 by 256-pixel MRI image [3].

4.3.1 Data Augmentation

As we mentioned before, one of the crucial challenges that all types of AI, machine learning, and deep learning models are facing is a lack of data sets, especially in medical imaging. Scientists spend too much time to make their algorithms very sophisticated to perform well. Unfortunately,

we can find many cases where this effort prevents them from paying enough attention to how their data sets should be. In this way, two significant problems may happen. First, their algorithms might memorize the training data sets instead of learning the features from given data [1]. Secondly, the model may learn very little or nothing from the data. The mentioned problems are caused by either lack of sufficient training data or too much uniformity of the data set.

The ultimate purpose of machine learning and deep learning models is to train a generalized model to correctly classify or estimate an input that is different from the training and testing data set. To reach this goal, we should try to escape from those two mentioned challenge.

The big motivation for data augmentation in medical imaging projects with deep learning is the lack of useful training data sets. The number of scanned images in the medical field, such as MRI, CT scans and sonograms, are way less than those produced as training data sets for standard image and video processing. Medical imaging is, first of all, costly, and patients are reluctant to be scanned for study purposes because they must be exposed to magnetic fields and in some cases specific chemicals. Moreover, strict privacy policies and patient's consent limits access to medical image repositories. A widely used method to avoid over-fitting and expanding data sets is data augmentation.

Data set augmentation in this project is done by randomly overturning the ordered chronology of the frames and feeding them alongside original sequences to the model. Also, we flip images against the horizontal axis for a random number of sequences. As another augmentation technique, we minimize the spatial dimension of the training data sets and randomly crop sequence of images in 352×352 pixel dimensions and applied a flip effect. All these augmentation actions are done to generalize the model and protect it from memorizing high-quality training set optical flows.

4.3.2 Refining Data set

Perez and Wang [33] proved that we could extract useful features even from low-quality images, which is very useful for training deep learning models. However, it does not mean that we should not care about outliers. Outliers, in many cases, have a destructive effect on data because machine

learning models are prone to extract false features from outliers. During data preparation, paying excessive attention to distinguishing between valuable data and outlier is necessary. Removing outliers from the data set highly improves the accuracy and performance of models. The practice of evaluating data augmentation on the accuracy of the model shows that data preparation is as important as designing the model [4].

As mentioned before, in our training data sets, thousands of MR images were masked by physicians to hide the identity of the patients, i.e., the eyes, nose or face part of MR images were blackened. These manual masks may mislead the training process because the black areas in consecutive sequences do not obey the same pattern and are drawn randomly.

4.4 Training our Neural Network

We trained the network with a different configuration from that suggested by Jiang et al. [18] to achieve an optimum result for MRI. First and foremost, we used the same data set this paper has suggested to see the performance of the model on MR images.

We split the data set into training, testing, and validation folders, each folder consisting of sequences of 12 consecutive temporal frames and spatial slices. As mentioned before, data augmentation was done to expand the data set. As the first augmentation method, in the data set level, the entire sequence was flipped horizontally and vertically and picked nine images in a row to train the network and evaluate for seven image interpolations. At the frame level, frames were resized to 360×360 pixels and cropped randomly to 352×352 pixels. The data augmentation process is done with a separate class of data manipulation.

The training ran for 500 epochs with the learning rate of 0.0001 which automatically decreased every 200 epochs by the factor of 10^{-1} to run from local minimums.

The second data set with which we trained the model was the Sunnybrook data set consist of cardiac, left and right ventricles. This data set consists of about 32,000 usable images for training this model, where we manually removed about 8,000 of those. From the remaining 25,207 images, we split the data set to train, validation and test with the proportions of 80%, 10%, and 10%. In

total, we used more than 19300 image sequences for training and more than 2400 image sequences for testing and the same number of image sequences for validation.

The network has been trained with the default Adam optimizer [20] for 700 epochs. We designed and implemented the mechanism of processing, converting and optimizing medical DICOM images for this network from scratch. The data preparation part of the original network was designed for video processing and could not be used for our application.

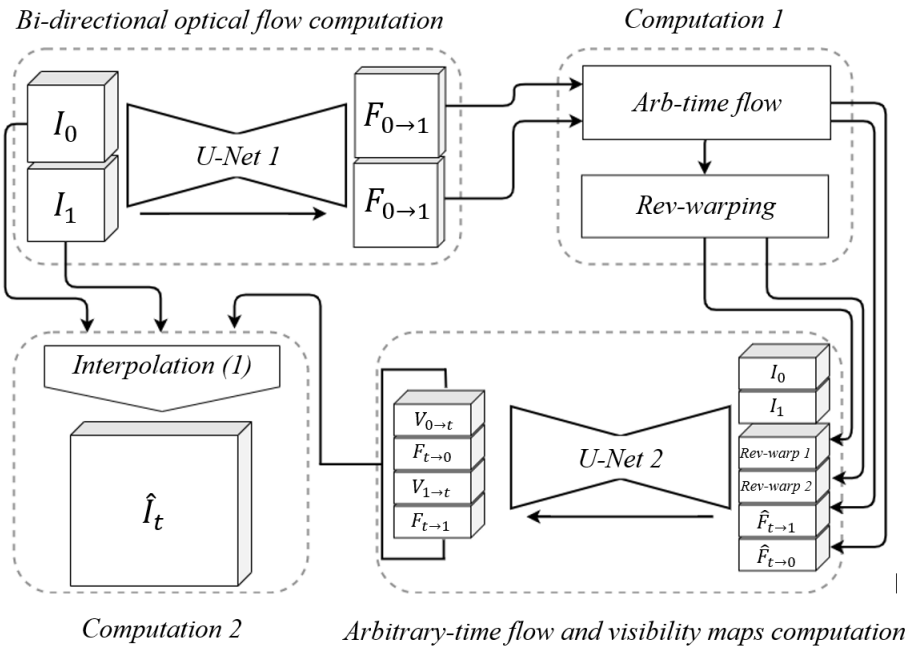

Figure 5: Flow diagram of the end-to-end network. The network receives two base images alongside the number of desired intermediate images and delivers synthetic images.

Finally, we used the Sunnybrook data set to fine-tune weights calculated by initial training on the Adobe 240-fps and YouTube 240-fps data sets. The benefit of transfer learning is that we specialize the weights, which were calculated by a massive number of MR images to our application.

4.5 Loss Functions

Jiang et al. [18] used a linear combination of multiple loss functions. Each of its parts contribute to improve the accuracy of the model in generation of intermediate frames or slices.

In Equation 5, I_0 and I_1 are the base frames base on which we are synthesizing intermediate images. $\left\{\hat{I}_{t_i}\right\}_{i=1}^{N}$ $(0 < t_i < 1)$ indicate synthesized (interpolated) images. $\{I_{t_i}\}_{i=1}^{N}$ $(0 < t_i < 1)$ are the ground-truth images. The loss function calculates the difference (error) between each I_{t_i} with corresponding $\hat{I}_{t_i}$ for the same arbitrary time t $(0 < t < 1)$. The linear combination of loss functions is defined as following Equation:

$$l = \lambda_r l_r + \lambda_p l_p + \lambda_\omega l_\omega + \lambda_s l_s \tag{5}$$

where l_r models how good is the interpolation of the same intermediate frame by computing the mean absolute difference of the pixels:

$$l_r = \frac{1}{N} \sum_{i=1}^{N} \left\| \hat{I_{t_i}} - I_{t_i} \right\| \tag{6}$$

The reconstruction loss formula is defined in the RGB space, in which pixel intensity values are in the range of 0 to 255. Using the mean absolute difference loss function might cause blurry synthetic images [19]. Imagine a synthetic image is highly similar to the original image but with one or two pixel offset relative to the original image. In this case, computing per-pixel difference (loss) might result in a significant error, although they are perceptually identical. So an additional loss function has been specifically trained by [19] to extract high-level features of an image. The following equation calculates the perceptual loss.

$$l_p = \frac{1}{N} \sum_{i=1}^{N} \left\| \left(\phi \hat{I_{t_i}} \right) - \left(\phi I_{t_i} \right) \right\| \tag{7}$$

where ϕ denotes the convolution ($conv4$-3) feature of the VGG-16 model trained on ImageNet [39].

To model the quality of warped images, the following terms were added to Equation 8. The first and second terms compute the reconstruction error of warped images by estimated flow vectors between two input images (I_0 and I_1). The third and the fourth terms compute the errors of the images which are warped by the bi-directional optical flow vectors ($\hat{F}_{t_0}$ and $\hat{F}_{t_1}$), where $\hat{F}_{0 \to t} =$

$t.F_{0\to1}$ and $\hat{F}_{1\to t} = -(1-t).F_{1\to0}$.

$$l_\omega = \|I_0 - g(I_1, F_{0\to1})\| + \|I_1 - g(I_0, F_{1\to0})\| + \frac{1}{N}\sum_{i=1}^{N}\left\|I_{t_i} - g(I_0, \hat{F}_{t_i\to0})\right\| + \frac{1}{N}\sum_{i=1}^{N}\left\|I_{t_i} - g(I_1, \hat{F}_{t_i\to1})\right\| \tag{8}$$

Smoothness loss (Equation 9) is the last part of the loss function. It forces neighboring pixels to have similar optical flow values [25, 18].

$$l_s = \|\Delta F_{0\to1}\| + \|\Delta F_{1\to0}\| \tag{9}$$

Jiang et al. [18] calculated the λ values empirically and set to $\lambda_r = 0.8$, $\lambda_p = 0.005$, $\lambda_\omega = 0.4$, and $\lambda_s = 1$ to achieve the best performance.

4.6 Training Tools and Resources

Programming languages and libraries

We used the Python programming language and several image processing, machine learning, and data analytic libraries including TensorFlow, TensorBoard, Keras, PyTorch, OpenCV, Pillow, PyPlot, and Pandas.

Server hardware specifications

For the training of the our CNN, we used a non-distributed server with the following specifications.

- Processor: Intel® Core™ i7 8700K, 3.70GHz with 6 cores and 12 threads.
- 64 gigabytes of DDR4 main main memory.
- NVIDIA® GeForce® GTX 2080 graphic processor with 8 gigabytes of memory.
- Solid state drive (Samsung 960 EVO NVMe) for faster and smoother saving and retrieval of data.

Cloud processing service

We used Google Colab as a distributed cloud GPU-enabled processing service [11].

5 Evaluation

In this section, we designed and evaluated multiple interpolation scenarios. To avoid any bias, we used a different evaluation data set than what we used for the training of the CNN. The evaluation data set consists of 10 slices with 6 millimeter spatial distance. Each slice was scanned every 100 milliseconds for 25 times, for a total of 250 images for evaluation.

For comparison of Fast-MRI with a related interpolation method, we interpolated images with the same configuration by Adaptive Separable Convolutions method (SepConv) that Niklaus et al. [31] proposed for image interpolation by deep learning.

5.1 Evaluation Methods

Every scientific research project needs an evaluation method to show the performance of the work. We used different methods to assess image synthesis quality. Wang and Bovik [42] categorized evaluation methods as full-reference and no-reference approaches. In full-reference methods, the original image and the distorted or synthesized image are both completely available for comparison. On the contrary, in no-reference or partial-reference evaluation, the original image is not available or partially available for comparison. In this work, since the scanned MR temporal or spatial images are available as reference, we used full-reference comparison.

5.1.1 Peak Signal-to-Noise Ratio

One of the popular metrics for image quality comparison in the field of image processing is Peak Signal-to-Noise (PSNR) [15]. PSNR is applied when an image reconstruction, i.e., compression or de-noising algorithm, needs to be assessed. This metric measures the ratio of the maximum possible value (MPV) of a signal, to the power of distorting noise (PDN). The unit of PSNR is decibel.

$$PSNR = 20log_{10}\left(\frac{Max_f}{\sqrt{MSE}}\right) \tag{10}$$

and the Mean Squared Error (MSE) in the above equation is:

$$MSE = \frac{1}{m.n} \sum_{i=1}^{m} \sum_{i=1}^{n} \|(f(i,j) - g(i,j)\|^2 \tag{11}$$

where:

- f is the original image matrix.
- g is the data as matrix form of the degraded or synthetic image.
- m is the width of the image.
- n is the height of the image.
- Max_f is the maximum intensity value in the original image.

5.1.2 Mean Squared Error

The most popular method which is used for image reconstruction quality assessment is Mean Squared Error (MSE). In this method, we calculate the square difference between the pixel intensities of scanned images and reconstructed or synthesized images. This metric is computationally efficient and easy to compute [21]. However, the problem with this method is that it does not give a sense of structural difference [10]. Synthesized images sometimes have different pixel intensities in comparison with the ground-truth, but they are structurally the same, and the quality is preserved. So, high Mean Squared Error does not always imply low reconstruction quality. We calculate the Mean Squared Error by the Equation 11.

5.1.3 Structural Similarity Index

Structural similarity index (SSIM) is a method for comprehensive measurement of degradation of the quality of images in luminance, contrast, and structure. This method relies on this assumption that the human visual system (HVS) has been adapted for extraction of visual structures [42]. The following Equation shows how to calculate SSIM.

$$SSIM(I_0, I_1) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{12}$$

where μ_x, μ_y, σ_x, σ_y, and σ_{xy} are local mean and standard deviations and cross covariance of images I_0 and I_1. $C_1 = (0.01.L)^2$, where L is the specified dynamic range value. $C_2 = (0.03.L)^2$, where L is the specifies dynamic range value and $C_3 = \frac{C_2}{2}$.

5.1.4 Pixel by Pixel Difference

All of the metrics mentioned above provide mathematical methods to compare signals and images. However, visual quality assessment by human in applications like image interpolation and reconstruction is crucial because the human visual system is designed and highly adapted to structural information of scenes and images [21].

Pixel by pixel comparison is a simple visual image comparison method. In the pixel by pixel comparison, we subtract the intensity values of the original from the reconstructed (or interpolated) image in the same pixel location. This method is very effective for the Human Visual System (HVS), especially for assessment of medical image reconstruction. Biologists use pixel by pixel difference as the primary tool to observe which part of the image reconstructed with higher quality. Mathematical evaluation methods, e.g., MSE, PSNR, and SSIM, accumulate errors of the pixels inside a region of interest and outside it. By having a visual representation to observe images, physicians can ignore the noise outside the regions of interest [14].

5.2 Evaluation Scenarios

For the evaluation of Fast-MRI, we designed five different scenarios to assess the performance of the model. To have an accurate and credible assessment, the spatial spacing distance (see Figure 6) between slices and the temporal time difference between frames in the evaluation data set must be known. Finding such a data set that has detailed information about the temporal and spatial distances is not an easy task. The data set that we used for evaluation is in DICOM format (see the documentation in [7]) and contains comprehensive meta-data including the temporal interval

in milliseconds, and spatial slice spacing in millimeters.

Temporal distance is the scanning time difference between two frames by MRI scanner in milliseconds. Think of it as the time interval between two frames of a video sequence. We produce videos or MR images by decreasing temporal difference of the frames (see Figure 7).

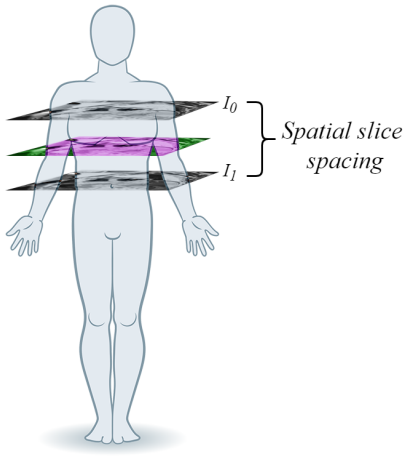

Figure 6: Spatial slice spacing. The distance between two scanned slices. A synthetic slice (marked in purple) is created by base slices (I_0 and I_1).

Slice spacing (Figure 6) is the spatial distance between two consecutive scanned MRI slices in millimeters. Increasing the distance between two slices in MRI produces less spatially coherent images because of drastic changes in the shape of the body organs.

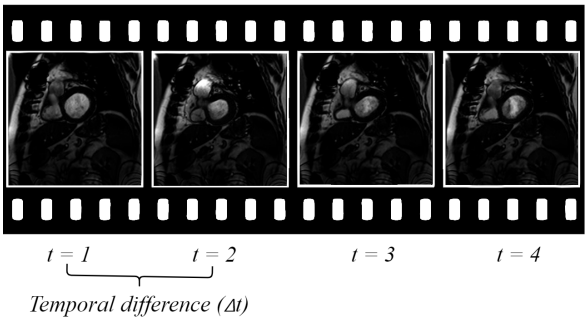

Figure 7: Temporal difference. The temporal difference between two base frames. The synthetic frames are generated according to these frames.

5.2.1 Temporal Single-Frame Synthesis

Given a sequence of scanned frames from a fixed slice of the body, the goal is to interpolate the intermediate frame $\hat{I}_t$, first based on two neighboring frames $I_{t-\Delta t}$ and $I_{t+\Delta t}$. Then we interpolate $\hat{I}_t$ based on the second level of neighboring frames $I_{t-2\Delta t}$ and $I_{t+2\Delta t}$, and we continue increasing the Δt by 100 milliseconds at each step for seven steps. The temporal differences of base frames for steps 1 to 7 are 200, 400, 800, 100, 1200, and 1400 milliseconds (see Figure 8). We repeat this evaluation scenario for temporal sequences scanned from 10 different cardiac slices. As we have the real scanned sequence of frames, we compared the interpolated frames ($\hat{I}_t$) with the ground-truth (I_t) and reported respective evaluation metrics discussed in Section 5.1.

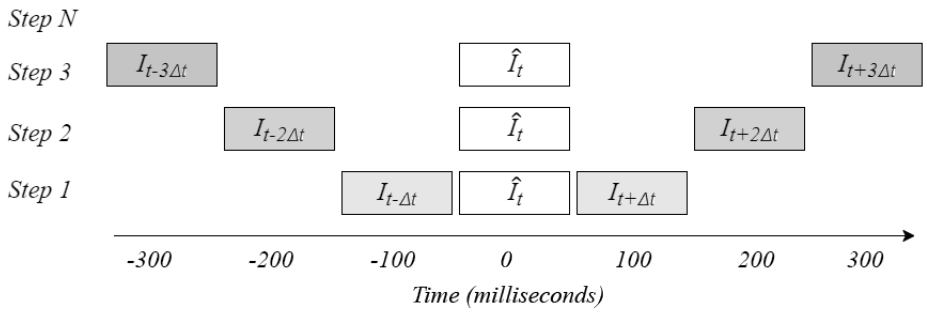

Figure 8: Temporal single-frame synthesis evaluation scenario. The evaluation process is performed for 7 neighboring pairs of base frames with temporal difference of 200-1400 milliseconds.

With the higher MRI scanning frame-rate, the calculated motion vectors of the body organs by optical flow in consecutive frames are more accurate, so we maintain the coherency assumption [13]. In this condition, we expect lower interpolation errors; on the contrary, we expect higher error and lower similarity for interpolated frames $\hat{I}_t$ based on based frames with higher temporal difference (lower scanning frame rates).

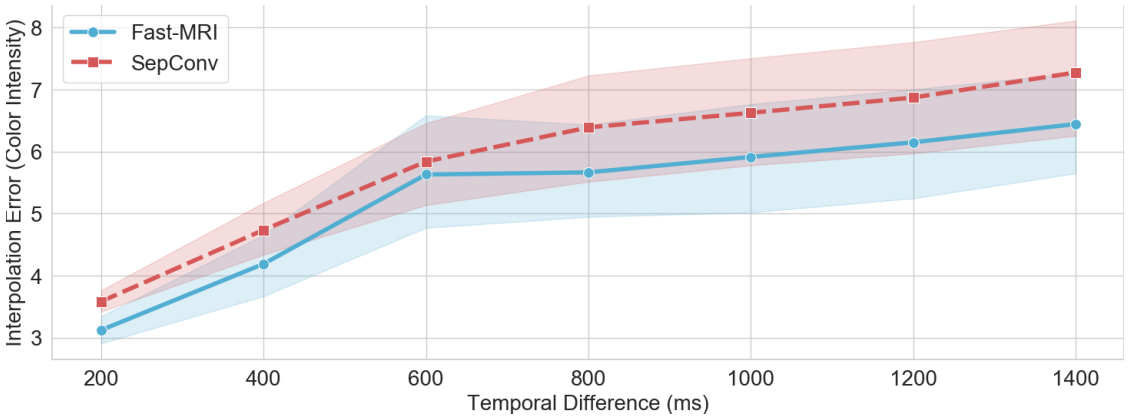

Figure 9: Temporal single-frame comparison results. The comparison between Fast-MRI and the SepConv shows Fast-MRI has less interpolation error on average. Increasing the temporal difference of the base frames increases the interpolation error.

Figure 9 shows increasing temporal differences between base frames increases the interpolation error for both Fast-MRI and SepConv. In Figure 10, red pixels are the synthesized pixels in the interpolated frame with different intensity than what they should have. By increasing the temporal distance of the base frames from 200 to 1400 milliseconds, the error is increased. The reason is that the frames with high temporal intervals are less coherent in the deformation caused by the motion of the heart. The stronger dark pixels in the bottom row in Figure 10 represent more interpolation errors (See Figure 23 in Appendix for detailed visual comparison of the results of Fast-MRI and SepConv).

Table 2 indicates that Fast-MRI outperforms SepConv with up to 12.9% less interpolation error. Increasing the temporal distance of the base frames decreased the PSNR values of both methods. This pattern is the same for the similarity (SSIM) of the interpolated frames and the ground-truth.

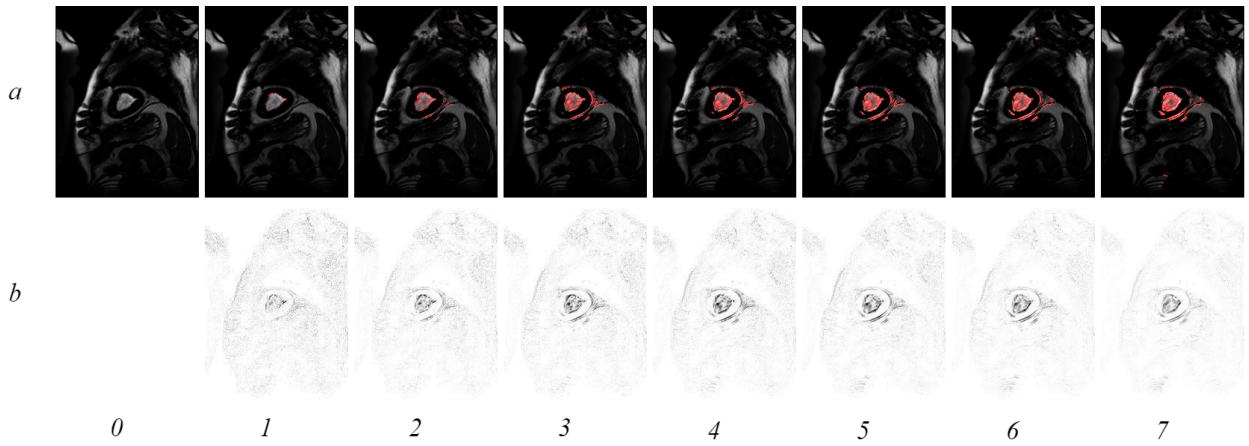

Figure 10: Temporal single-frame visual comparison. (a.0) is the ground-truth frame with which in every step, we compared the interpolated frame. (a.1)-(a.7) are $\hat{I}_t$ based on $I_{t-\Delta t}$ and $I_{t+\Delta t}$ for Δt of 100-700 milliseconds, show increasing temporal difference of two base frames which decreases the similarity index and increases the interpolation error for the synthetic images. Pixel differences are marked in red. (b.1)-(b.7) depict inverted normalized subtraction of ground-truth and synthetic images. Dark pixels shows the difference between the interpolated frames and ground-truth.

Table 2: The results of temporal single-frame synthesis.

Temporal single-frame synthesis	Average Interpolation Error (color intensity)		Average PSNR * (dB)		Average SSIM ** (x100 %)	
	Method					
Temporal difference (milliseconds)	SepConv	Fast-MRI	SepConv	Fast-MRI	SepConv	Fast-MRI
	(Lower is better)		(Higher is better)		(Higher is better)	
200	3.58	**3.12**	37.98	**38.30**	0.96	**0.97**
400	4.73	**4.19**	35.14	**35.84**	0.95	**0.96**
600	5.84	**5.63**	33.12	**33.44**	0.94	**0.94**
800	6.39	**5.66**	32.28	**33.33**	0.94	**0.94**
1000	6.62	**5.91**	31.92	**32.95**	0.93	**0.94**
1200	6.87	**6.15**	31.57	**32.61**	0.93	**0.94**
1400	7.28	**6.44**	31.01	**32.18**	0.92	**0.93**
* PSNR: Peak signal to noise ratio. ** SSIM: Structural similarity index.						

5.2.2 Spatial Single-Slice Synthesis

Assuming we are given a sequence of cardiac slices scanned in a specific time from multiple parts of the heart. Fast-MRI synthesizes intermediate slice based on the next neighboring slices, then based on next level of the neighboring slices and so on (see Figure 11). The slice spacings of the based slices for this scenario are 12, 24, 36, and 48 mm. We compared the interpolated slices ($\hat{I}_s$) with the ground-truth (I_s) to assess the performance of the model for spatial single-slice interpolation.

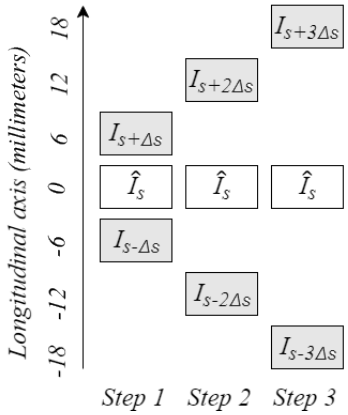

Figure 11: Spatial single-slice synthesis evaluation scenario. The evaluation process was performed for 3 neighboring pairs of base frames with slice spacings of 12, 24, 36, and 48 millimeters.

The organs of the human body have different shapes and conditions. Similarly, the shape of the heart changes in different spatial locations. If we want to have good interpolation results, we should select two base slices with the least possible spatial spacing, because the rate of the changes in the shape of the subject, i.e., heart, kidney, and vessels are considerably high in the spatial domain.

Figure 12 visualizes that increasing the cardiac base slices spacing increases the interpolation error (Section 5.1.3). Compared with SepConv, Fast-MRI has a lower error in all of the slice spacing.

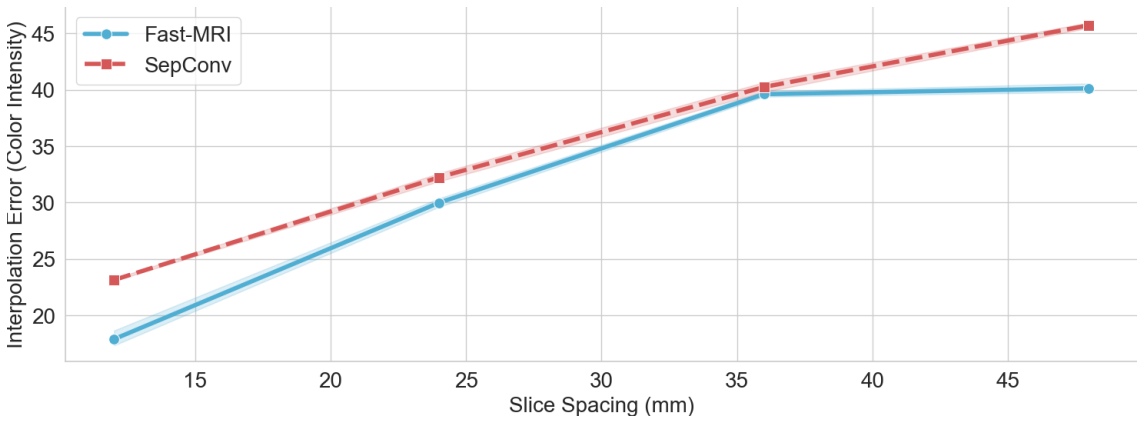

Figure 12: Spatial single-slice comparison results. The comparison of Fast-MRI with SepConv. Although the interpolation error is higher for base slices with larger slice spacing, Fast-MRI outperforms SepConv in all of the evaluated spatial spacing.

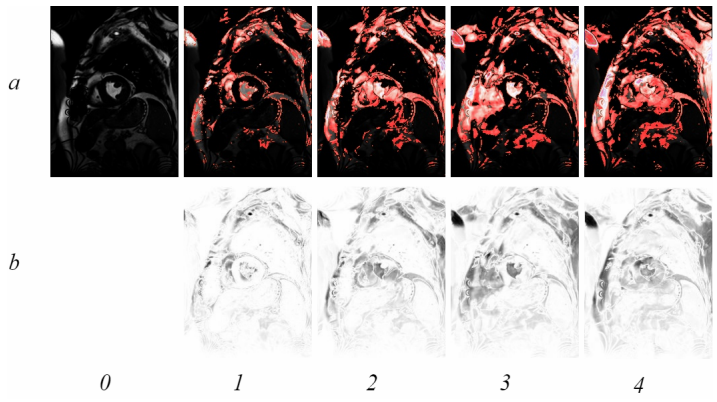

Figure 13: Spatial single-slice visual comparison. (a.0) represents the ground-truth slice with which in every step, we are trying to synthesize and compare. (a.1)-(a.4) $\hat{I}_s$ based on $I_{s-\Delta s}$ and $I_{s+\Delta s}$ for Δs of 6, 12, 18, and 24 millimeters. (b.1) visualizes normalized inverted subtraction of ground-truth and synthetic intermediate slice. Increasing the spatial spacing of the base slices causes increase in interpolation error.

Figure 13 visualizes interpolation errors of the interpolated slices with Fast-MRI (see detailed visual comparison of Fast-MRI with SepConv in Figure 24 in Appendix).

Table 3 shows Fast-MRI outperforms SepConv by 10.8% decrease in average interpolation error. The errors of both methods are higher in this scenario compared with the previous scenario because the input images are not coherent for base slices with 12-48 mm of spacing. Similarity index and PSNR of the interpolated slices and the ground-truth are inversely related to the base slice spacing.

Table 3: The results of spatial single-slice synthesis.

Spatial single-slice synthesis	Average Interpolation Error (color intensity)		Average PSNR * (dB)		Average SSIM ** (x100 %)	
	Method					
Slice spacing (millimeters)	SepConv	Fast-MRI	SepConv	Fast-MRI	SepConv	Fast-MRI
	(Lower is better)		(Higher is better)		(Higher is better)	
12	23.13	**17.88**	21.96	**23.12**	0.58	**0.65**
24	32.23	**29.98**	18.26	**18.60**	0.41	**0.43**
36	40.24	**39.61**	15.94	**16.18**	0.33	**0.32**
48	45.72	**40.13**	14.65	**16.06**	0.25	**0.31**
* PSNR: Peak signal to noise ratio. ** SSIM: Structural similarity index.						

5.2.3 Temporal Multi-Frame Synthesis

In the third scenario (see Figure 14), we evaluated the performance of the Fast-MRI in synthesis of multiple intermediate temporal frames based on two neighboring MR frames. Given a sequence of 9 consecutive temporal frames I_1 to I_9, we interpolated:

1. Frame $\hat{I}_5$ based on I_4 and I_6 based frames.
2. Frames $\hat{I}_4$, $\hat{I}_5$ and $\hat{I}_6$ bases on I_3, and I_7.
3. Frames $\hat{I}_3$, $\hat{I}_4$, $\hat{I}_5$, I_6, and I_7 based on I_2, and I_8.
4. Frames $\hat{I}_2$, $\hat{I}_3$, $\hat{I}_4$, $\hat{I}_5$, $\hat{I}_6$, $\hat{I}_7$, and $\hat{I}_8$ based on I_1 and I_9.

The temporal difference between each frame is 100 ms, so the time intervals between above-mentioned base frames are 200, 400, and 800 ms.

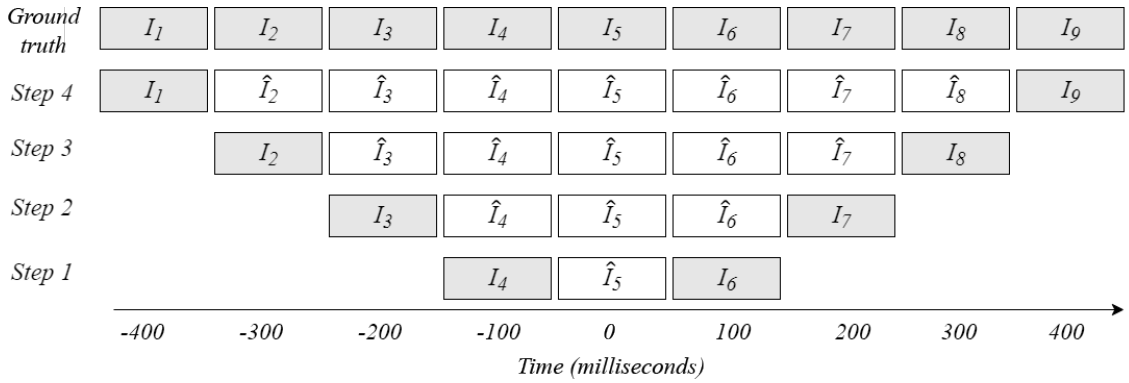

Figure 14: Temporal multi-frame synthesis evaluation scenario. The evaluation process was performed for two base frames with temporal differences of 200-800 milliseconds. Gray-filled rectangles indicate base frames. White rectangles represent synthesized in-between frames.

Figure 15 shows the comparison of Fast-MRI with the related method in interpolation error. The frames closer to the base frames have lower errors because of the higher contribution of one of base frames according to Equation 1. Fast-MRI outperforms the proposed method by Niklaus et al. [31].

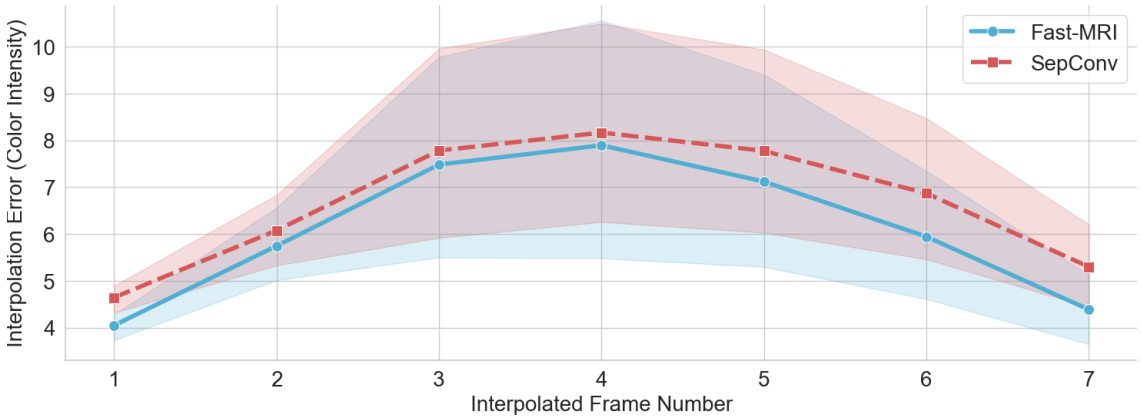

Figure 15: Temporal multi-frame comparison results. The comparison between Fast-MRI and SepConv. Fast-MRI has less error in temporal multi-frame interpolation scenario.

Table 4 shows that Fast-MRI has higher performance in interpolation of multiple-frames in temporal MRI, with a 3.3%-17.2% decrease in interpolation error. Visualizing pixel differences of Fast-MRI gives better understanding of the performance because the similarity index in Fast-MRI

Table 4: The results of temporal multi-frame synthesis.

Temporal multi-frame synthesis		Average Interpolation Error (color intensity)		Average PSNR * (dB)		Average SSIM ** (x100 %)	
		Method					
Number of interpolations	Slice number in the sequence	SepConv	Fast-MRI	SepConv	Fast-MRI	SepConv	Fast-MRI
		(Lower is better)		(Higher is better)		(Higher is better)	
1	1	3.88	**3.49**	36.95	**37.39**	0.96	**0.97**
3	1	4.83	**4.12**	34.76	**35.92**	0.96	**0.96**
	2	5.44	**4.94**	33.62	**34.37**	0.95	**0.95**
	3	7.43	**4.38**	31.15	**35.52**	0.93	**0.96**
5	1	-	4.24	-	35.67	-	0.96
	2	-	5.68	-	33.18	-	0.95
	3	-	6.65	-	32.22	-	0.94
	4	-	5.92	-	33.30	-	0.95
	5	-	4.05	-	36.30	-	0.96
7	1	4.64	**4.05**	35.06	**36.04**	0.95	**0.96**
	2	6.07	**5.74**	32.55	**33.13**	0.94	**0.94**
	3	7.78	**7.48**	30.7	**31.37**	0.93	**0.93**
	4	8.16	**7.89**	30.44	**31.11**	0.93	**0.93**
	5	7.77	**7.12**	30.73	**31.88**	0.93	**0.94**
	6	6.87	**5.94**	31.74	**33.18**	0.94	**0.95**
	7	5.30	**4.39**	34.05	**35.60**	0.95	**0.96**

* PSNR: Peak signal to noise ratio.
** SSIM: Structural similarity index.
- Scenario for five frame synthesis was not possible by SepConv.

is higher (see Figure 16).

Most of the pixels that contain error and have different intensity values compared with the ground-truth are located in the middle part of the frame. This is where the moving part of the heart is located. Motionless areas (pixels near edges of the frame) are not expected to have considerable differences (see Figure 16). The contribution of the base frames is very high for the neighboring synthetic frames (Equation 1) thus the error is lower than the interpolated frames in the middle of the sequence.

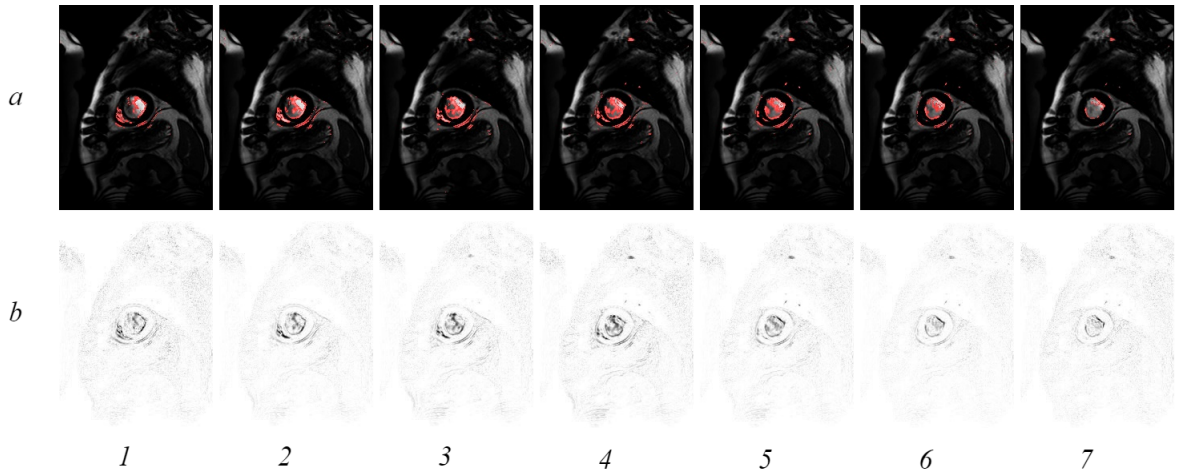

Figure 16: Temporal multi-frame visual comparison. (a.1)-(a.7) are comparisons of the multiple interpolated frames and the respective ground-truth. The error in the synthetic frames located in the middle of the sequence is higher.(b.1)-(b.7) visualizes the normalized inverted subtraction of the synthesized frames from the respective ground-truth frame. Dark pixels show errors. See detailed visual comparison of the Fast-MRI and SepConv in Figure 25 in Appendix).

5.2.4 Spatial Multi-Slice Synthesis

In this scenario (see Figure 17) we assessed the performance of Fast-MRI for spatial multiple intermediate slice synthesis. For a sequence of 9 consecutive slices, we generated synthetic slices in the following order.

1. Slice $\hat{I}_5$ based on I_4 and I_6 based slices.
2. Slices $\hat{I}_4$, $\hat{I}_5$, and $\hat{I}_6$ bases on I_3 and I_7.
3. Slices $\hat{I}_3$, $\hat{I}_4$, $\hat{I}_5$, $\hat{I}_6$, and $\hat{I}_7$ based on I_2, and I_8.
4. Slices $\hat{I}_2$, $\hat{I}_3$, $\hat{I}_4$, $\hat{I}_5$, $\hat{I}_6$, $\hat{I}_7$, and $\hat{I}_8$ based on I_1 and I_9.

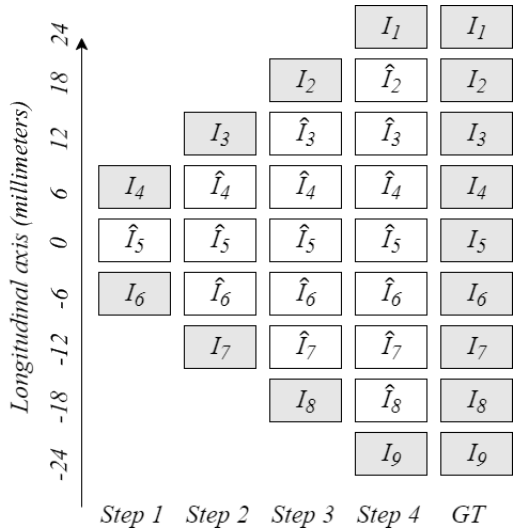

Figure 17: Spatial multi-slice synthesis evaluation scenario. The evaluation process was performed for 12 to 48 millimeters of spatial spacing between base slices. Grey-filled rectangles represent available scanned ground-truth frames. White rectangles represent synthesized in-between slices.

The spacings of base slices are 12, 24, 36, and 48 mm for 1, 3, 5, and 7 multi-slice interpolation, respectively. We compared each interpolated slice with its corresponding ground-truth image. We also performed the same procedure for SepConv; however, because this method generates only a single slice in-between two base slices, evaluation for 5 slices cannot be done by this method.

Figure 18 represents the comparison of the interpolation errors. The synthesized slices closer to the base slices have less error as we expected. By moving from the first and last slices in the sequence to the middle slices, the error is increased (see Section 5.2.3).

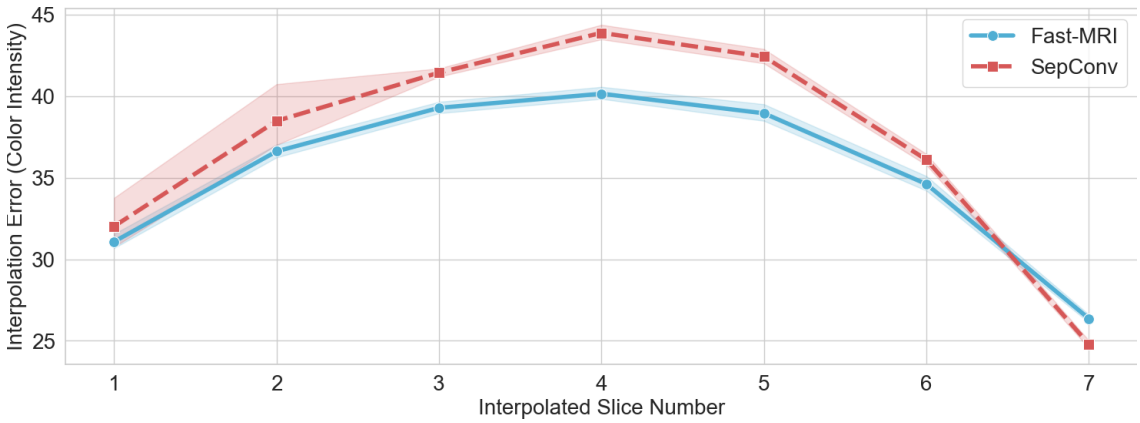

Figure 18: Spatial multi-slice comparison results. Fast-MRI outperforms SepConv in interpolation error. The error is higher in the synthetic slice in the middle of the interpolated sequence.

As highlighted in Table 5, Fast-MRI outperforms SepConv by at least 2.9%. Because of the high spatial spacing and accordingly low coherency of the spatial slices, the error values are higher for temporal multi-frame synthesis. The structural similarity index and peak signal-to-noise-ratio of the synthesized images at the beginning and end of the interpolated sequence is higher than synthetic images in the middle of the sequence for the same reason mentioned earlier.

Figure 19 represents pixel by pixel differences of the interpolated slices by Fast-MRI with ground-truth. The error is higher compared with temporal multi-frame interpolation because of higher base slice distances and lower coherency (check Figure 26 in Appendix for detailed visual comparison of the slices interpolated by Fast-MRI and SepConv).

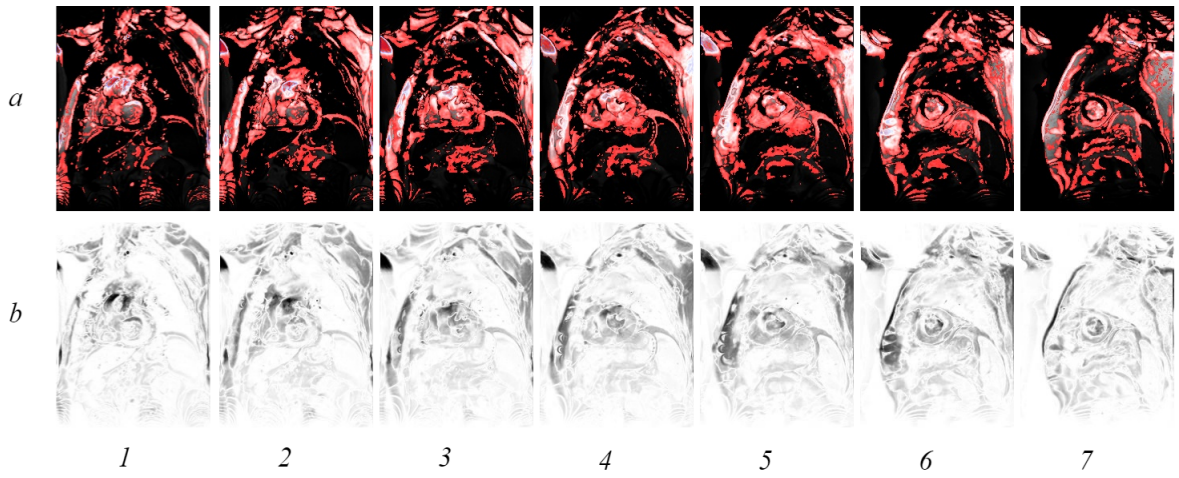

Figure 19: Spatial multi-slice visual comparison. (a.1)-(a.7) colorizes the pixel difference of the synthesized slices compared to their respective ground-truth. (b.1)-(b.7) visualizes the normalized inverted subtraction of the synthesized slices from the respective ground-truth. (a.4) The slice farthest from the base images has the highest error because of the less contribution of base slices.

Table 5: The results of spatial multi-slice synthesis.

Spatial multi-slice synthesis		Average Interpolation Error (color intensity)		Average PSNR * (dB)		Average SSIM ** (x100 %)	
		Method					
Number of interpolations	Slice number in the sequence	SepConv	Fast-MRI	SepConv	Fast-MRI	SepConv	Fast-MRI
		(Lower is better)		(Higher is better)		(Higher is better)	
1	1	-	19.72	-	22.23	-	0.62
3	1	**31.39**	33.05	**18.27**	17.76	**0.41**	0.40
	2	**33.69**	34.39	**17.66**	17.41	0.36	**0.36**
	3	36.9	**31.35**	16.9	**18.22**	0.34	**0.42**
5	1	-	35.3	-	17.18	-	0.38
	2	-	37.71	-	16.61	-	0.3
	3	-	38.1	-	16.51	-	0.29
	4	-	37.69	-	16.61	-	0.34
	5	-	31.19	-	18.25	-	0.43
7	1	32.02	**31.08**	18.1	**18.29**	0.39	**0.41**
	2	38.48	**36.61**	16.58	**16.86**	0.31	**0.31**
	3	41.44	**39.28**	15.94	**16.25**	0.28	**0.28**
	4	43.88	**40.15**	15.46	**16.06**	0.29	**0.31**
	5	42.42	**38.95**	15.74	**16.33**	0.33	**0.33**
	6	36.1	**34.61**	17.08	**17.35**	0.36	**0.36**
	7	**24.79**	26.36	**20.2**	19.71	0.47	**0.47**

* PSNR: Peak signal to noise ratio.
** SSIM: Structural similarity index.
- Scenario for five frame synthesis was not possible by SepConv.

5.2.5 Coherency Challenge

Frames should be coherent spatially and temporarily; otherwise, the model is not able to map corresponding pixels to each other and calculate correct optical flow. Furthermore, combining warped frames of two non-coherent MRI slices or frames (Equation 2) results in an inaccurate interpolated image. In medical imaging, this phenomenon happens when:

- Two consecutive slices have high spacing, which means they are spatially far from each other.
- Two consecutive frames have a large temporal difference, which means the scanning interval of the scanner was set to a high number. Also, if we scan a very fast-moving organ in the body (e.g., heart), even with relatively low intervals, maybe we face non-coherency issues.
- One or more synthetic slices or frames may be affected by motion artifacts. By dropping the imperfect slices/frames, the model cannot reconstruct reliable in-between images based on nearest motion artifact-free images.

In Figure 20, the synthesized intermediate slice is not quantitatively and visually similar to the ground-truth. The large error in the results is because of non-coherency caused by high spatial spacing between two based slices (48 mm). Those slices are not structurally similar and represents two parts of the heart with completely different shapes and sizes.

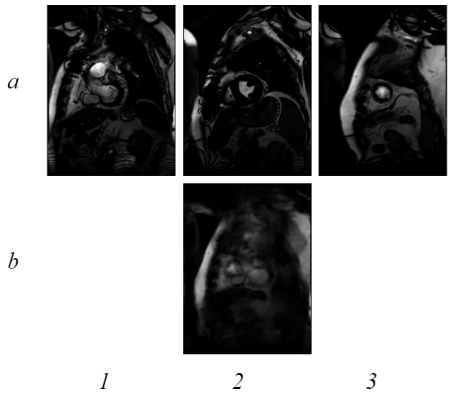

Figure 20: Coherency problem. (a.1) and (a.3) are base frames with spatial spacing of 48 millimeters. (a.2) is the ground-truth image we are trying to reconstruct. (b.1) is the synthetic image based on (a.0) and (a.2). Because of low spatial coherency between (a.1) and (a.3), the result is not visually and structurally similar to the ground-truth and the interpolation error is high.

5.2.6 Frame Synthesis Based on Synthesized Frames

In this evaluation scenario, similar to Section 5.2.2, we interpolate a single slice but by using two previously synthesized frames as base slices. The error is expected to be accumulated because each interpolated frame of slice contains a certain amount of error. Also, we expect lower similarity for interpolated images compared with single-slice interpolation by real scanned based slices.

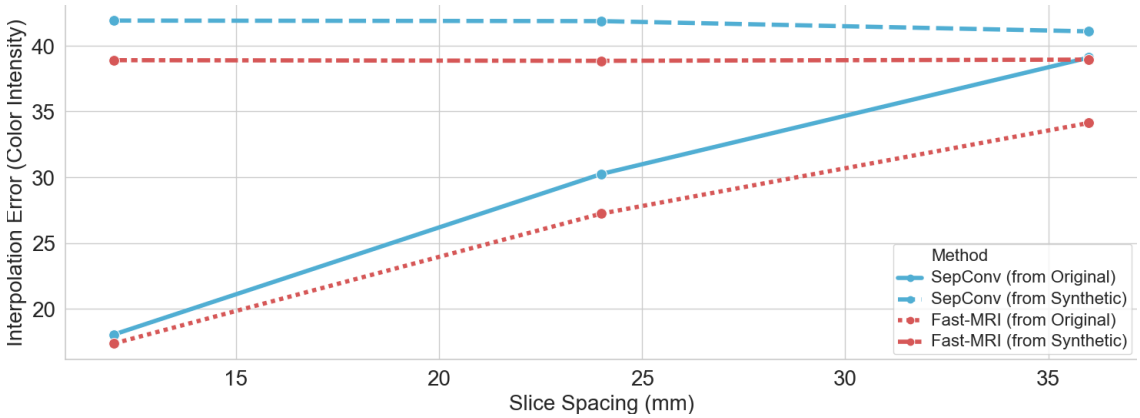

Figure 21: Evaluation of slice synthesis based on synthetic slices. Comparison of single slice synthesis based on real base slices vs. synthesis base slices. The increasing lines indicate a larger error for high slice spacing. The constant lines in the upper section of the image show very high error of interpolation based on synthetic images. In both scenarios, Fast-MRI outperforms SepConv.

From Figure 21, we can observe that by increasing the spacing of the base slices, the interpolation error of the slices that are synthesized by the real images are gradually increased. However, for the slices synthesized by synthetic slices, the error is high from the beginning. Figure 22 shows that even for small base slice spacing (12 mm), the results have very low similarity and high error. Despite the low performance of the both methods in this scenario, Fast-MRI outperforms SepConv with an up to 7.2% lower error (see Table 6). The output MR images are not visually satisfying and the left and right ventricles of the heart are not recognizable.

Table 6: The results of slice interpolation based on synthetic slices.

Interpolation based on synthetic	Interpolation Error (color intensity)		PSNR * (dB)		SSIM ** (x100 %)	
Slice spacing distance (milliseconds)	Fast-MRI	SepConv	Fast-MRI	SepConv	Fast-MRI	SepConv
	(Lower is better)		(Higher is better)		(Higher is better)	
12	**38.88**	41.89	**16.33**	15.68	**0.34**	0.31
24	**38.84**	41.85	**16.34**	15.69	**0.32**	0.30
36	**38.93**	41.07	**16.32**	15.85	**0.35**	0.32
* PSNR: Peak signal to noise ratio. ** SSIM: Structural similarity index.						

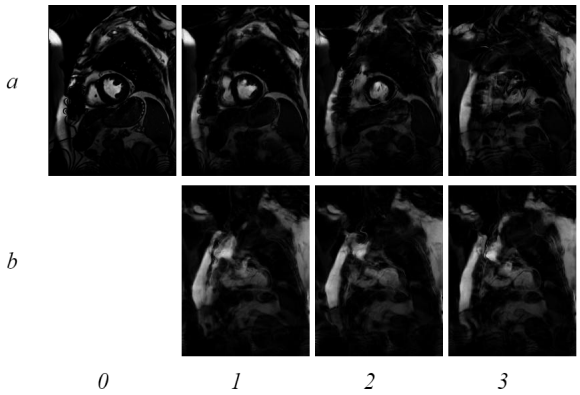

Figure 22: Visual comparison of slice synthesis based on synthetic slices. (a.0) is the ground-truth slice. (a.1)-(a.3) are the synthetic slices based on real slices with 12, 24, and 36 millimeters of spacing. (b.1)-(b.3) are the synthetic slices based on synthetic slices with the same spacing similar to row (a) base images. The similarity of the synthetic based interpolation is low and the error, even in 12 millimeters of spacing, is higher than in synthesis by real slices.

5.2.7 CINE MRI Video Frame Synthesis

Cine MRI is a short video sequence that shows the flow of cardiac or cerebrospinal fluid during a single heartbeat. Radiologists can measure fluid flow inside the heart or brain of a patient and compare it with a healthy subject to diagnose blockage in vessels [5]. As a visual evaluation scenario, we took a low frame rate CINE MRI and interpolated in-between frames to increase the frame rate of the sequences. The motion of the bloodstream is smoother in the interpolated sequence by Fast-MRI (the comparison of interpolated CINE videos are uploaded online [6]).

6 Conclusion

In this thesis, we redesigned, trained and evaluated an end-to-end network specifically to generate arbitrary number of intermediate MR images. By the first convolutional neural network (CNN), we estimated motion vectors between input images. By the second CNN, we estimated the visibility maps to address occlusion problem and refined flow vectors to produce intermediate images. We used MR images alongside high frame-rate non-MR video frames to train the the networks. We designed five evaluation scenarios to comprehensively cover temporal and spatial, single and multiple-frame interpolation on a specific evaluation data set. Our proposed method (Fast-MRI) outperforms one of the accessible single-frame interpolation related methods [31] in all of the designer evaluation scenarios. For future work, we will improve the performance of the work by using larger MRI data set. Also we will integrate this work to work on Microsoft's HoloLens.

Bibliography

[1] *An Introduction to Optical Flow.* www.youtube.com/watch?v=hBMMmsaWWHwn. Accessed: 2019-07-13.

[2] John L. Barron, David J. Fleet, and Steven S. Beauchemin. "Performance of Optical Flow Techniques". In: *International Journal of Computer Vision* 12.1 (1994), pp. 43–77.

[3] *Cardiac MRI Dataset.* medicine.utah.edu/radiology/news/2017/07/sms_cardiac_mri.php. Accessed: 2019-10-10.

[4] Davide Chicco. "Ten Quick Tips for Machine Learning in Computational Biology". In: *BioData Mining* 10.1 (2017), p. 35.

[5] *CINE MRI, The American Syringomyelia Chiari Alliance Projec.* asap.org/index.php/disorders/cine-mri/. Accessed: 2019-08-13.

[6] *CINE results by Fast-MRI.* youtu.be/UL36xrCPM6k. Accessed: 2019-11-11.

[7] *DICOM Standard Overview.* www.dicomstandard.org/about/. Accessed: 2019-11-02.

[8] Alexy Dosovitskiy et al. "Learning Optical Flow with Convolutional Networks". In: *Proceedings of the IEEE International Conference on Computer Vision.* 2015, pp. 2758–2766.

[9] Kunihiko Fukushima. "Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position". In: *Biological Cybernetics* 36.4 (1980), pp. 193–202.

[10] Bernd Girod. "What's Wrong With Mean-squared Error?" In: *Digital Images and Human Vision* (1993), pp. 207–220.

[11] *Google Colab - Colaboratory.* colab.research.google.com/. Accessed: 2019-06-11.

[12] Evan Herbst, Steve Seitz, and Simon Baker. "Occlusion Reasoning for Temporal Interpolation Using Optical Flow". In: *Department of Computer Science and Engineering, University of Washington, Tech. Rep. UW-CSE-09-08-01* (2009).

[13] Berthold K.P. Horn and Brian G. Schunck. "Determining Optical Flow". In: *Artificial Intelligence* 17.1-3 (1981), pp. 185–203.

[14] *How Image Comparison Works?* `support.smartbear.com/testcomplete/docs/testing-with/checkpoints/regions/how-image-comparison-works.html`. Accessed: 2019-08-14.

[15] Quan Huynh-Thu and Mohammed Ghanbari. "Scope of Validity of PSNR in Image/video Quality Assessment". In: *Electronics Letters* 44.13 (2008), pp. 800–801.

[16] Eddy Ilg et al. "Flownet 2.0: Evolution of Optical Flow Estimation with Deep Networks". In: *Proceedings of the Ieee Conference on Computer Vision and Pattern Recognition*. 2017, pp. 2462–2470.

[17] J. Yu Jason, Adam W. Harley, and Konstantinos G. Derpanis. "Back to Basics: Unsupervised Learning of Optical Flow via Brightness Constancy and Motion Smoothness". In: *European Conference on Computer Vision*. Springer. Amsterdam, The Netherlands, 2016, pp. 3–10.

[18] Huaizu Jiang et al. "Super Slomo: High Quality Estimation of Multiple Intermediate Frames for Video Interpolation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Salt Lake City, UT, USA, 2018, pp. 9000–9008.

[19] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. "Perceptual Losses for Real-time Style Transfer and Super-resolution". In: *European Conference on Computer Vision*. Springer. Amsterdam, The Netherlands, 2016, pp. 694–711.

[20] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[21] Yuming Li et al. "No-reference Image Quality Assessment With Shearlet Transform and Deep Neural Networks". In: *Neurocomputing* 154 (2015), pp. 94–109.

[22] Xiaodan Liang et al. "Dual Motion GAN for Future-flow Embedded Video Prediction". In: *proceedings of the IEEE International Conference on Computer Vision*. Venice, Italy, 2017, pp. 1744–1752.

[23] Qinghua Lin, Qin Zhang, and Li Tongbin. "Slice Interpolation in Mri Using a Decomposition-reconstruction Method". In: *4th International Conference on Information Science and Control Engineering (ICISCE)*. IEEE. Changsha, China, 2017, pp. 678–681.

[24] Geert Litjens et al. "A Survey on Deep Learning in Medical Image Analysis". In: *Medical Image Analysis* 42 (2017), pp. 60–88.

[25] Ziwei Liu et al. "Video Frame Synthesis Using Deep Voxel Flow". In: *Proceedings of the IEEE International Conference on Computer Vision*. Venice, Italy, 2017, pp. 4463–4471.

[26] S-CB Lo et al. "Artificial Convolution Neural Network Techniques and Applications for Lung Nodule Detection". In: *IEEE Transactions on Medical Imaging* 14.4 (1995), pp. 711–718.

[27] Michael Lustig et al. "Compressed Sensing MRI". In: *IEEE Signal Processing Magazine* 25.2 (2008), p. 72.

[28] Dhruv Mahajan et al. "Moving Gradients: a Path-based Method for Plausible Image Interpolation". In: *ACM Transactions on Graphics (TOG)*. Vol. 28. 3. ACM. 2009, p. 42.

[29] Simone Meyer et al. "Phase-based Frame Interpolation for Video". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Boston, MA, USA, 2015, pp. 1410–1418.

[30] Simon Niklaus, Long Mai, and Feng Liu. "Video Frame Interpolation via Adaptive Convolution". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Venice, Italy, 2017, pp. 670–679.

[31] Simon Niklaus, Long Mai, and Feng Liu. "Video Frame Interpolation via Adaptive Separable Convolution". In: *Proceedings of the IEEE International Conference on Computer Vision*. Venice, Italy, 2017, pp. 261–270.

[32] *Overcoming One of The Biggest MR Imaging Challenges . . . scan time.* `newsroom.gehealthcare.com/overcoming-one-of-the-biggest-mr-imaging-challenges-scan-time-rsna/`. Accessed: 2019-06-14.

[33] Luis Perez and Jason Wang. "The Effectiveness of Data Augmentation in Image Classification Using Deep Learning". In: *arXiv preprint arXiv:1712.04621* (2017).

[34] Klaas P. Pruessmann et al. "SENSE: Sensitivity Encoding for Fast MRI". In: *Magnetic Resonance in Medicine* 42.5 (1999), pp. 952–962.

[35] Jerome Revaud et al. "Epicflow: Edge-preserving Interpolation of Correspondences for Optical Flow". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Boston, MA, USA, 2015, pp. 1164–1172.

[36] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer. Munich, Germany, 2015, pp. 234–241.

[37] Payam Samadi-Miyandoab, Ahmad Esmaili-Torshabi, and Saber Nankali. "2D and 3D Optical Flow Based Interpolation of the 4DCT ImageSequences in the External Beam Radiotherapy". In: *Frontiers in Biomedical Technologies* 2.2 (2015), pp. 93–102.

[38] Hoo-Chang Shin et al. "Stacked Autoencoders for Unsupervised Feature Learning and Multiple Organ Detection in a Pilot Study Using 4D Patient Data". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.8 (2012), pp. 1930–1943.

[39] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-scale Image Recognition". In: *arXiv preprint arXiv:1409.1556* (2014).

[40] *The basics of MRI*. `www.cis.rit.edu/htbooks/mri/inside.htm`. Accessed: 2019-11-01.

[41] *Wall Motion Research, Penn Medicine*. `www.pennmedicine.org/departments-and-centers/department-of-radiology/radiology-research/labs-and-centers/quantitative/cardiovascular-research-group/wall-motion`. Accessed: 2019-06-19.

[42] Zhou Wang and Alan C. Bovik. "Mean Squared Error: Love It or Leave It? A New Look at Signal Fidelity Measures". In: *IEEE Signal Processing Magazine* 26.1 (2009), pp. 98–117.

[43] Jiachen Zhuo and Rao P. Gullapalli. “MR Artifacts, Safety, and Quality Control”. In: *Radiographics* 26.1 (2006), pp. 275–297.

Appendix

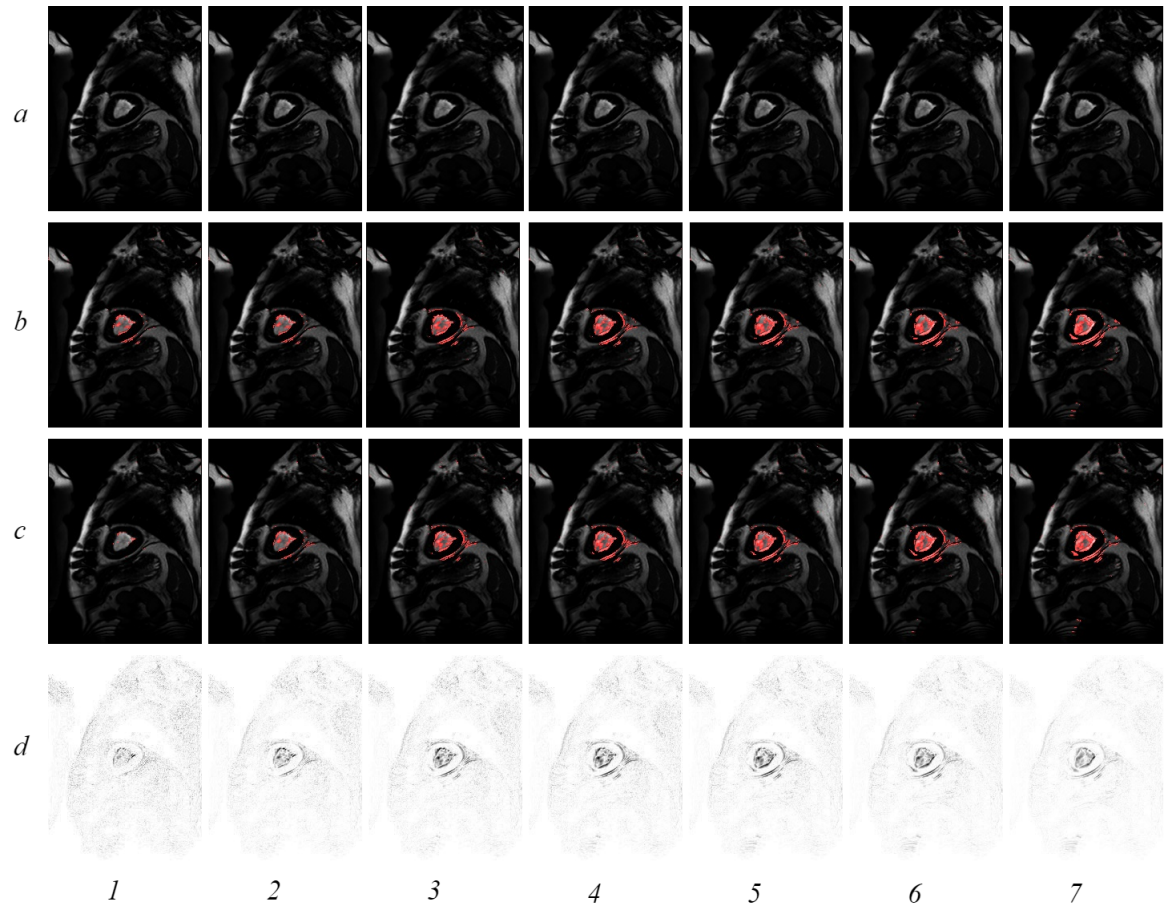

Figure 23: Comparison of temporal single frame generation. (a.1)-(a.7) Represent ground-truth, a single frame which our and related method generated frames with its timestamp. (b.1)-(b.7) synthesized frames by related method, based on base frames with temporal difference of 200, 400, 600, 800, 1000, 1200, and 1400 milliseconds. (c.1)-(c.7) synthesized frames by Fast-MRI, based on base frames with above mentioned temporal difference. (d.1)-(d.7) are inverted normalized subtractions of the ground-truth and respective frame generated by Fast-MRI.

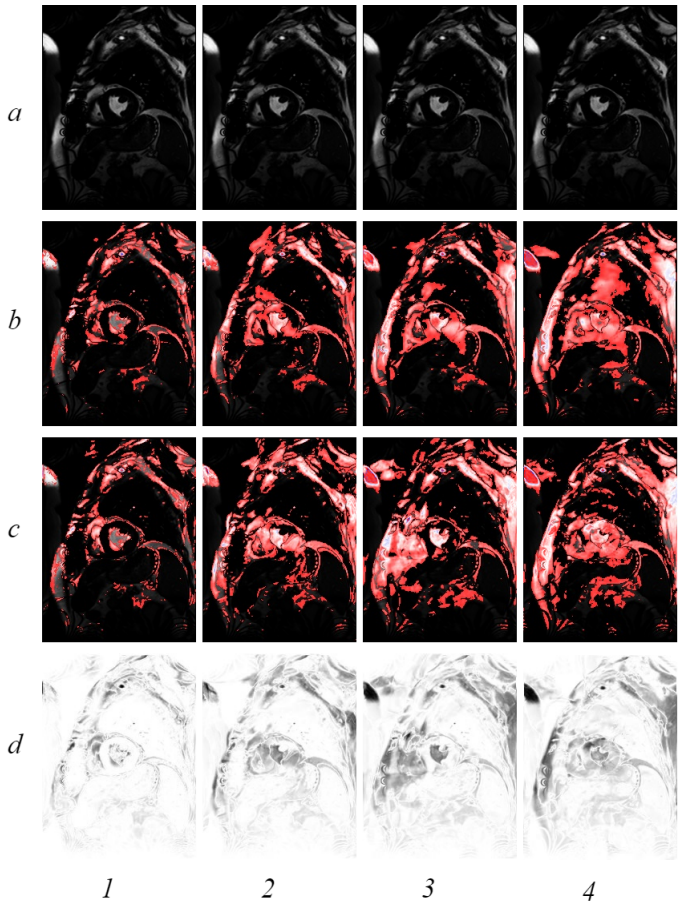

Figure 24: Comparison of Spatial single slice generation. (a.1)-(a.7) Represent ground-truth, a single slice which our and related method generated frames in its spatial location. (b.1)-(b.7) synthesized slices by related method, based on base slices with spatial spacing of 12, 24, 36, and 48 millimeters. (c.1)-(c.7) synthesized slices by Fast-MRI, based on base slices with the above mentioned spatial spacing. (d.1)-(d.7) are inverted normalized subtraction of the ground-truth and respective slices generated by Fast-MRI.

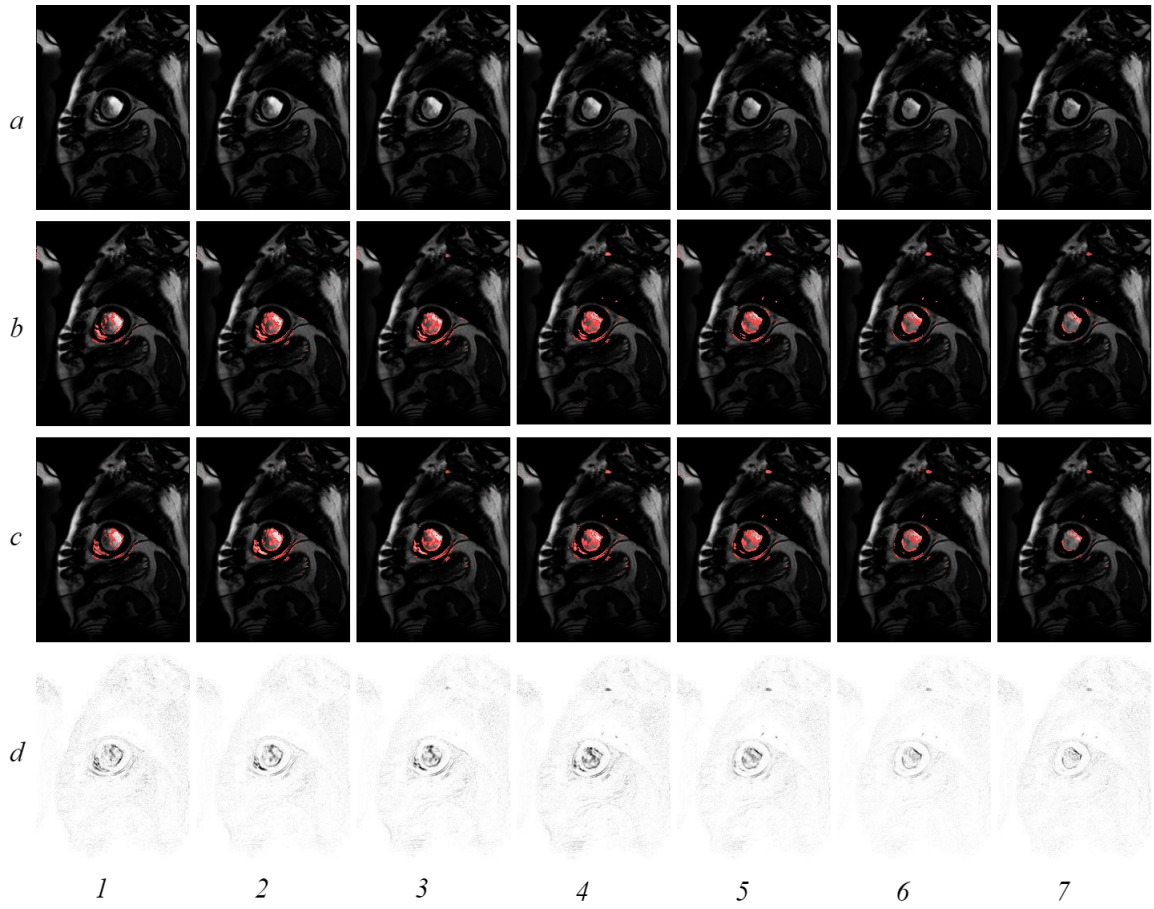

Figure 25: Comparison of temporal multi frame generation. (a.1)-(a.7) are ground-truth of a consecutive series of intermediate frames with a temporal difference of 100 milliseconds. (b.1)-(b.7) are multiple synthesized frames by related method in the exact time of the respective ground-truth. (c.1)-(c.7) are multiple synthesized frames by Fast-MRI in the exact time of the respective ground-truth. (d.1)-(d.7) are inverted normalized subtraction of the ground-truth and the respective frames generated by Fast-MRI.

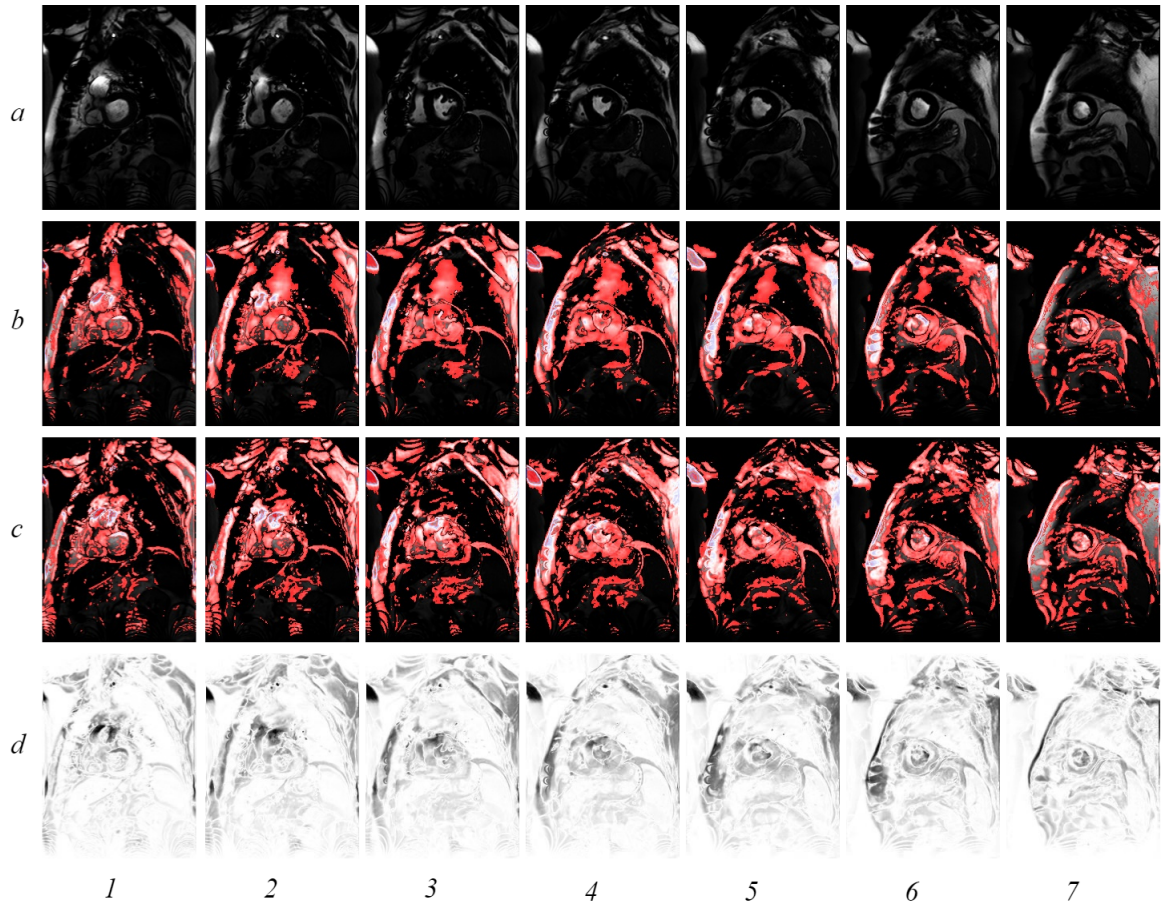

Figure 26: Comparison of Spatial multi slice generation. (a.1)-(a.7) are ground-truth of a consecutive series of intermediate slices with spatial spacing of 6 millimeters. (b.1)-(b.7) are multiple synthesized slices by related method in the exact same location of the respective ground-truth. (c.1)-(c.7) are multiple synthesized slices by Fast-MRI in the exact same location of the respective ground-truth. (d.1)-(d.7) are inverted normalized subtraction of the ground-truth and the respective slices generated by Fast-MRI.