FOAL: Fast Online Adaptive Learning for Cardiac Motion Estimation

Hanchao Yu*[2], Shanhui Sun†[1], Haichao Yu[2], Xiao Chen[1],
Honghui Shi†[3], Thomas Huang[2], Terrence Chen[1]
[1]United Imaging Intelligence, Cambridge, MA 02140, [2]University of Illinois at Urbana-Champaign, [3]University of Oregon
† shanhui.sun@united-imaging.com, † shihonghui3@gmail.com

Abstract

Motion estimation of cardiac MRI videos is crucial for the evaluation of human heart anatomy and function. Recent researches show promising results with deep learning-based methods. In clinical deployment, however, they suffer dramatic performance drops due to mismatched distributions between training and testing datasets, commonly encountered in the clinical environment. On the other hand, it is arguably impossible to collect all representative datasets and to train a universal tracker before deployment. In this context, we proposed a novel fast online adaptive learning (FOAL) framework: an online gradient descent based optimizer that is optimized by a meta-learner. The meta-learner enables the online optimizer to perform a fast and robust adaptation. We evaluated our method through extensive experiments on two public clinical datasets. The results showed the superior performance of FOAL in accuracy compared to the offline-trained tracking method. On average, the FOAL took only 0.4 *second per video for online optimization.*

1. Introduction

Video dense tracking and motion estimation using deep learning has gained great progress for natural image applications in recent research [35, 12, 22, 46, 49, 16, 11, 41, 18, 27, 51, 21]. In medical imaging, videos, compared to static images, are ideal for dynamically changing physiological processes such as the beating heart and are commonly used in clinical settings. Feature tracking of dynamic cardiac images can provide precise and comprehensive assessments of the cardiac motion and has been proved valuable for cardiac disease management [34, 28, 44, 24]. Motion estimation can also benefit other tasks in cardiac imaging, such as image reconstruction [10, 31] and semi-supervised segmentation [26, 38, 50, 17, 45, 42]. Recently, deep learning-based methods show promising results in cardiac motion estimation [26, 50, 15, 23]. However, most studies have been designed in a research environment: the proposed models are trained and tested on the data with similar distributions. In a clinical environment, however, the imaged objects may present various anatomies (abnormally thin or thick heart muscle) and/or dynamics (irregularly beating heart) for different diseases. On top of that, the imaging process itself commonly introduces many, if not more, variations. This is especially true for cardiac magnetic resonance (CMR) imaging, which provides superior video quality over ultrasound, but the image appearances are influenced by multiple factors including scanner vendors, main magnetic fields, different scanning protocols and technicians' operations. It is arguably impossible to build a dataset that includes every combination of the variations and train a universal tracker on it. It is also not ideal and sometimes impossible in a clinical setting that the pre-trained network gets fine-tuned on the data from a different distribution, given the scarce nature of medical data. In other words, for a clinically suitable deep-trained tracker, the neural network needs to possess the capability to quickly adapt to new data from unseen distributions. Towards this end, we propose a fast online adaptive learning (FOAL) mechanism for dense video tracking applied to cardiac motion estimation. The proposed framework consists of an online adaptive stage and an offline meta-learning stage. The offline meta-learning trains the model to gain the adaptation capability and the online stage will apply this adaptation to adjust the model parameters using very few and unseen data. We have designed a unique module for video tracking used in both stages to train an adaptive tracker. The tracker trained using the proposed FOAL achieves the state-of-the-art (SOTA) results compared to strong baselines. The contributions of our work are summarized as follows.

- In the context of dense motion estimation, we proposed a novel online model adaptation method, which adapts a trained baseline model to a new video using a gradient descent optimization.

- We proposed a meta-learning method optimizing the proposed online optimizer. The meta-learner enables

[*] This work was carried out during the internship of the author at United Imaging Intelligence, Cambridge, MA 02140.

the online optimizer to perform a fast and robust adaptation.

- We proposed practical solutions for training meta learner in dense motion estimation task.
- Our proposed method is not limited to the network structure of the baseline dense motion estimation. The extensive experiments consistently demonstrated superior performance improvement of our method in accuracy comparing to the baseline model.

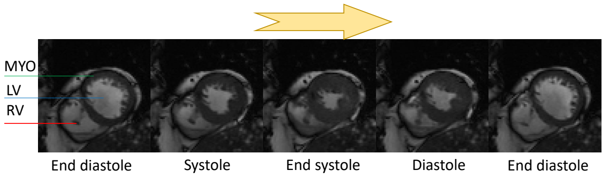

Figure 1. A typical cardiac cycle of a healthy subject recorded by CMR. The cycle indicates the heart relaxation and contraction process. The myocardium (MYO) appears as a dark ring in the image. The left ventricle (LV) is filled with a hyperintense blood signal contained inside the ring. The right ventricle (RV) cavity is indicated via the red line.

2. Related Work

Section 2.1 discusses state-of-the-arts in the literature for motion estimation in the computer vision field. Section 2.2 introduces the task of cardiac motion estimation and existing studies on this topic. Section 2.3 introduces the model-agnostic meta-learning which has inspired our method.

2.1. Motion Estimation for Camera Videos

Motion estimation is one of the fundamental problems in the computer vision field. In the literature, there are a few deep learning-based approaches solving motion estimation such as reported works in [2, 5, 12, 35, 22]. Dosovitskiy *et al*. [5] proposed two optical flow estimation networks (Flownets): FlownetSimple and FlownetCorr. The former is a generic architecture and the latter includes a correlation layer to fuse feature vectors at different image locations. Flownet 2.0 in the work [12] further adds an extra branch to deal with pairs with small displacement and uses the original Flownet to deal with large displacement. Sun *et al*. [35] proposed a smaller and more efficient neural network structure utilizing feature pyramid as well as cost volume to get a more accurate motion. Most of these above works used a supervised learning approach with true motion fields. In contrast to these supervised methods, Meister *et al*. [22] proposed an unsupervised framework where the flow was predicted and used to warp the source image to the reference image. The model is optimized to minimize the difference between the warped image and the reference image. Besides, an occlusion-aware forward-backward consistency loss is used with the census transform to improve the tracking results. Note that our baseline model utilized a similar self-supervision idea as [22].

2.2. Cardiac Motion Estimation

Cardiac motion estimation takes a time series (video) of CMR images as input and predicts the heart motion through time. Motion fields are usually estimated at a pixel level due to the non-rigid nature of cardiac contraction. Normally the video records a complete cardiac contraction cycle: from the onset of contraction (end-diastolic ED), then to maximum contraction (end-systolic ES) and back to relaxation. Fig. 1 shows example CMR frames from a video of a normal subject. The motion of a frame is usually estimated relative to a reference frame that is commonly chosen as the ED or ES frame. Let frame at time t be $I(x, y, t)$, and $I(x, y, t_{ref})$ as the reference image. The goal of motion estimation is to find the mapping F_θ such that

$$F_\theta : (I(x, y, t_{ref}), I(x, y, t)) \longrightarrow V_x(x, y, t), V_y(x, y, t) \tag{1}$$

where F_θ is the mapping function with parameter θ and V_x, V_y are the motion fields along x and y directions, respectively. Motion tracking methods can be generally categorized according to different formulations of F_θ: optical flow based, conventional image registration based, and deep learning based.

The optical flow based method is built on several presumptions on image appearance and motion strength, such as brightness consistency and small motion between the source and reference frames. The problem of applying optical flow based methods to CMR motion estimation is that the presumptions are violated in CMR videos [6]. Fig. 2 shows some example images, illustrating the challenges of CMR. Wang *et al*. [40] proposed a novel gradient-flow based method that uses a local shape model to keep the local intensity and shape features invariance.

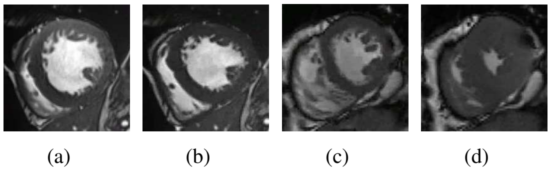

Figure 2. Examples of challenges in CMR motion estimation. (a) and (b) are from one CMR video, where the upper part of the LV myocardium (anterior wall) has a big intensity drop due to the changes in MR coil detection sensitivity. (c) and (d) are from another CMR video, where large motion occurs between an end-diastolic frame (source) and an end-systolic frame (reference).

In addition to the optical flow based approaches, image registration based methods [25, 29, 4, 32, 33, 37, 15, 47] were applied to solve cardiac motion estimation. Craene *et al*. [4] utilized B-spline velocity fields with physical constraints to compute the trajectories of feature points and performed the tracking. Rueckert *et al*. [29] proposed a free form deformation (FFD) method solving a general deformable image registration problem and recent work [25, 32, 33, 37] utilize this method to estimate the cardiac motion. It is known that FFD-like methods suffer from the computation efficiency problem. To address this issue, Vigneault *et al*. [39] proposed a coarse-to-fine registration framework to track cardiac boundary points. This solution improved the time efficiency but an extra segmentation step was required. In addition, this sparse tracking lost motion understanding in the heart muscle region.

Recent success in deep neural network solving many computer vision problems has inspired efforts to explore deep learning based cardiac motion estimation. Qin *et al*. [26] proposed a multi-task framework that combines segmentation and motion estimation tasks. The learned cardiac motion field is used to warp the segmentation mask and guide the segmentation module in a semi-supervised manner. The results show that both segmentation and motion estimation performance is improved compared to a single task. Zheng *et al*. [50] proposed the apparent flow net which is a modified U-net. The segmentation masks were used in the apparent flow net to improve motion estimation. In work [15], a conditional variational autoencoder (VAE) based method was presented to estimate the cardiac motion. The VAE encoder is used to map deformations to latent variables, which is regularized via Gaussian distribution and decode to a deformation filed via VAE decoder. Note that it is generally hard to obtain true cardiac motion and thus above works were quantitatively evaluated using the segmentation masks. In this work, we also use this type of evaluation.

2.3. Model Agnostic Meta Learning

Meta-learning, or learning to learn, aims to build a universal meta-model that could make fast adaptation to new tasks [30]. Model-agnostic meta-learning (MAML) [7] is a general strategy that searches for good model-agnostic initialization parameters that are trained through training tasks and can quickly adapt to new tasks. Given the initial model parameters θ, for every task T_i in the training set, the task-specific parameters θ_i are independently updated within the task dataset using gradient descent with a differentiable loss function L:

$$\theta_i \leftarrow \theta - \alpha \nabla_\theta L(T_i; \theta). \tag{2}$$

Then the original model parameters θ are updated over all the training tasks:

$$\theta \leftarrow \theta - \beta \nabla_\theta \sum_i L(T_i; \theta_i). \tag{3}$$

Through these meta-training processes, the optimal "initialization" parameters are supposed to be sensitive to new task adaptation within a limited number of adaptation steps. MAML has been widely used in few-shot learning [8, 36, 9], neural architecture search [20], graphical neural network [9], compressed sensing [43] and transfer learning [48]. Most applications using MAML are to solve high-level vision tasks such as classification and recognition. The MAML method inspired us to utilize a meta learner which teaches the model to learn how to adapt to a new video.

3. Method

We proposed an online adaptive tracking framework in the context of dense motion tracking utilizing a deep neural network. The proposed method is a general video tracking framework that is not limited to motion estimation in CMR. Nevertheless, without loss of generality, the method is presented in the CMR context.

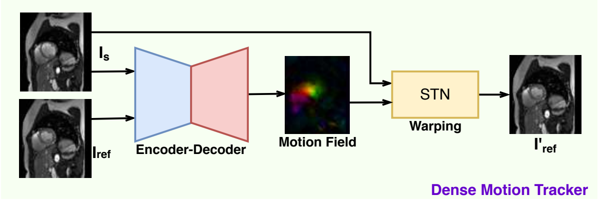

Figure 3. Overview of the dense tracking framework. The encoder is a Siamese structure that takes source and reference images as input. The feature maps produced by the Siamese encoder are concatenated and fed into the decoder.

3.1. Dense Motion Tracking

Fig. 3 depicts the architecture of our dense tracking framework. The overall idea of the dense motion tracking is an end-to-end unsupervised learning approach that inspired from [22]. Annotating the motion field for the heart is an intractable task and unsupervised learning avoids the necessity of the ground truth. In our work we used a lightweight backbone of the network: the inputs are source image and reference image (e.g. two frames in the same video). The encoder is a Siamese [3] structure. The decoder is a series of convolution and transpose convolution operators used to decode the features and restore the output to the original image size. The output is the predicted motion field. To perform unsupervised learning, the spatial transformer network [13] is utilized to deform/warp the source image to the reference image and image reconstruction loss L_{mse} is used to minimize the difference between the warped source image and the reference image. L_{mse} is the mean square error (MSE). In addition to L_{mse}, motion field smoothness L_{smooth} proposed in [26] is used to avoid abrupt motion change and a bidirectional (forward-backward) flow consistency loss L_{con} proposed in [22] is used. The total loss

L_{total} is thus defined as follows:

$$L_{total} = L_{mse} + \alpha_s L_{smooth} + \beta_c L_{con}, \quad (4)$$

where α_s and β_c are used to balance the three losses.

3.2. Online Optimizer

The unsupervised dense tracking (Section 3.1) mitigates the need for ground truth motion fields. However, the distribution mismatch between training and test datasets is a continuous challenge, particularly the long tail problem in the medical image domain. The clinical deployment of a deep learning model suffers the domain mismatch problem. It is a challenge to collect sufficient samples to train a universal tracker. In this section, in the context of the proposed dense tracking, we extend the tracker to address the dataset distribution mismatch problem. Instead of training such a universal tracker offline, we make the tracker being aware of the test data online. The idea behind this is to enable a given tracker to automatically adapt to a new video x. Suppose we have a model f_θ using the proposed dense tracker trained on dataset D_a with a distribution $p(D_a)$. The online adaptive learning on video x is an online optimization algorithm and is realized via back-propagating through the stochastic gradient descent steps as follows:

$$\theta' \leftarrow \theta' - \alpha \nabla_{\theta'} L(f_{\theta'}), \quad (5)$$

where θ' represents the model parameters and is initialized from θ. α is the learning rate. We utilized the same loss function L defined in Eq.(4). The overview of the online adaptive algorithm is outlined in the Algorithm 1.

Algorithm 1 FOAL online optimization

```
Input: Single video $k$: $x_k$, learning rate: $\alpha$, trained model:
  $f_\theta$, number of online tracking optimization steps: $M$
  $\theta'_k \leftarrow \theta$
  Sample $K$ pairs $D_k = \{a_k^{(j)}, b_k^{(j)}\}$ from video $x_k$
  for $m$ from 1 to $M$ do
      Evaluate loss $L_m(f_{\theta'_k})$ using $D_k$
      Compute parameters with gradient descent:
      $\theta'_k \leftarrow \theta'_k - \alpha \nabla_{\theta'_k} L_m(f_{\theta'_k})$
  end for
Output: updated network weights: $\theta'_k$
```

It is worth pointing out that the gradient descent steps are performed over all parameters of the network at the online stage. Thus, it is computationally expensive to optimize them on all image pairs (source and reference) with too many steps. We aim to adapt the offline model in just a few steps using only a small number of online samples. We realize this by utilizing meta-learning to optimize this optimization procedure. This idea is inspired by MAML [7], which is used to learn good initial model parameters via meta-learning. Like in MAML, we perform a second-order optimization by back-propagation using stochastic gradient descent through the online optimization Eq. (5).

Algorithm 2 FOAL offline meta-learning

```
Input: video set: $X$, learning rate: $\alpha$, $\beta$, initial model: $f_\theta$,
  number of online tracking optimization steps: $m$
  while not done do
      Sample $N$ videos $\{x_1, x_2, ..., x_N\}$ from $X$
      for $i$ from 1 to $N$ do
          $\theta'_i \leftarrow \theta$
          Sample $K$ pairs $D_i = \{a_i^{(j)}, b_i^{(j)}\}$ from video $x_i$
          for $t$ from 1 to $m$ do
              Evaluate loss $L_i(f_{\theta'_i})$ using $D_i$
              Compute parameters with gradient descent:
              $\theta'_i \leftarrow \theta'_i - \alpha \nabla_{\theta'_i} L_i(f_{\theta'_i})$
          end for
          Sample $K$ pairs $D'_i = \{a_i^{(k)}, b_i^{(k)}\}$ from video $x_i$
      end for
      Model update: $\theta \leftarrow \theta - \beta \nabla_\theta \frac{1}{N} \sum_i^N L_i(f_{\theta'_i})$ using
  each $D'_i$ and video-specific loss $L_i(f_{\theta'_i})$
  end while
Output: updated model $\theta$
```

3.3. Meta-learning

We utilized a meta leaner to re-train the model f_θ on the dataset D_{meta} from parameters θ in order to teach the online optimizer in Eq. (5). The optimizer learns to adapt f_θ for a given video x. Note that D_{meta} is either $p(D_a)$ or a new distribution $p(D_b)$, where D_b is a new dataset, and $p(D_b)$ may mismatch domain $p(D_a)$. The full algorithm is outlined in Algorithm 2. There are two For-loops in Algorithm 2. The inner For-loop is the proposed optimization algorithm in Algorithm 1 to optimize the online optimizer Eq. (5). The outer For-loop is the meta-leaner and the meta optimizer is defined as follows.

$$\theta \leftarrow \theta - \beta \nabla_\theta \frac{1}{N} \sum_i^N L_i(f_{\theta'_i}), \quad (6)$$

where i is i^{th} video in the training procedure. N is the number of videos in a batch size for optimizing the meta learner. β is the learning rate of the meta-learner. L_i is the loss (Eq. 4) evaluated on the i^{th} video. $f_{\theta'_i}$ is the model parameters for the i^{th} video.

3.4. Practical Version of the Meta-Learning

Memory limitation and solution: In contrast to few-shot learning (a classification problem) discussed in MAML [7], dense motion tracker need store a larger number of feature maps (i.e. requiring a large amount of GPU memory) given a larger image size (e.g. 192 × 192). The

Table 1. Inside distribution v.s. outside distribution Dice coefficient results for the baseline model, proposed FOAL without meta-learning (FOAL w/o meta) and proposed FOAL with meta-learning (FOAL + meta). Averaged Dice coefficient with standard deviation is given among five-fold leave-one-disease-out cross-validation.

Method	LV	RV	MYO
	Inside Distribution Test Set		
Baseline	0.838(0.024)	0.825(0.013)	0.797(0.014)
FOAL w/o meta	0.856(0.021)	0.842(0.013)	0.820(0.008)
FOAL + meta	**0.873(0.019)**	**0.859(0.013)**	**0.840(0.007)**
	Outside Distribution Test Set		
Baseline	0.840(0.094)	0.775(0.096)	0.803(0.045)
FOAL w/o meta	0.863(0.077)	0.801(0.085)	0.828(0.031)
FOAL + meta	**0.880(0.065)**	**0.806(0.086)**	**0.846(0.027)**

Table 2. Inside distribution v.s. outside distribution Hausdorff distance (mm) results for the baseline model, proposed FOAL without meta-learning (FOAL w/o meta) and proposed FOAL with meta-learning (FOAL + meta). Averaged Hausdorff distance with standard deviation is given among five-fold leave-one-disease-out cross-validation.

Method	LV	RV	MYO
	Inside Distribution Test Set		
Baseline	7.265(0.779)	8.782(0.422)	6.930(0.548)
FOAL w/o meta	6.417(0.627)	8.141(0.329)	6.286(0.469)
FOAL + meta	**6.012(0.580)**	**7.731(0.303)**	**6.157(0.489)**
	Outside Distribution Test Set		
Baseline	6.921(2.147)	10.173(1.436)	6.716(1.803)
FOAL w/o meta	6.158(1.727)	9.320(1.422)	6.107(1.506)
FOAL + meta	**5.832(1.534)**	**9.378(1.417)**	**5.987(1.437)**

meta optimizer (Eq. 6) requires computing derivatives of each independent model associated with a specific video. To tackle this problem, by employing the property that the gradient operator and the average operator are commutative in Eq. 6, we swap the two operators as shown in Eq. (7).

$$\nabla_\theta \frac{1}{N}\sum_{i}^{N} L_i(f_{\theta'_i}) \Leftrightarrow \frac{1}{N}\sum_{i}^{N} \nabla_\theta L_i(f_{\theta'_i}) \qquad (7)$$

which enables computing gradients on GPU and transferring them to CPU.

First order derivative approximation: Note that in Eq. (7), second-order derivative is needed in back-propagation. This involves calculating the second-order Hessian matrix, which is computationally costly. As a workaround, we use first-order approximation, whose effectiveness is demonstrated in MAML [7]. In [7], the approximation rendered comparable results to the second-order derivatives.

4. Evaluation Methodology

In this section, we present evaluation methodology on compared tracking methods: tracking performed using proposed dense motion tracking method (baseline model), tracking performed using online optimization from the baseline model without meta-learning (FOAL without meta-learning), and tracking performed using online optimization with meta-learning (FOAL with meta-learning).

4.1. Datasets and Evaluation Reference

In our study, two public CMR datasets were utilized: ACDC dataset [1] and Kaggle Data Science Bowl Cardiac Challenge Data [14]. All data acquisitions were performed using breath-holding so that only cardiac motion is observed in the videos. It is arguably impossible to make an independent reference standard of the cardiac motion manually. To perform quantitative analysis, we utilized segmentation masks as the independent reference standard. In the test dataset of the study, we have heart segmentation references at both the first frame and the evaluated reference frame. We generate the segmentation masks via warping source segmentation to the reference and compare it to the annotation using quantitative indices defined in section 4.4.

ACDC Dataset: It includes short-axis view CMR videos from 100 subjects (healthy and diseased cases). Each subject contains multiple slices (9-10) and each slice is a video sequence covering at least one heartbeat cycle. Overall, there are 951 videos in this dataset. Each video provides two heart segmentation masks: one for the ED phase and one for the ES phase. The segmentation labels are right ventricle (RV) cavity, myocardium (MYO) and left ventricle (LV) cavity. In addition, the 100 subjects are evenly divided into 5 categories with 20 subjects each. These are diagnosed into: normal cases (NOR), systolic heart failure with infarction (MINF), dilated cardiomyopathy (DCM), hypertrophic cardiomyopathy (HCM), abnormal right ventricle (ARV). The CMR videos were collected over 6 years using two MRI scanners of different main magnetic fields: 1.5 T Siemens Area and 3.0 T Siemens Trio Tim (Siemens Medical Solutions, Germany) [1].

Kaggle Data Science Bowl Cardiac Challenge Dataset: It includes short-axis view CMR videos from 1100 subjects. Each subject contains multiple slices (8-10) and each slice is a video sequence covering at least one cardiac cycle. Overall, there are 11202 videos in this dataset. The original challenge is to predict the ejection fraction from the videos. Ejection fraction ground truth was provided but irrelevant to our study. The subjects have a large health and age range and the images were collected from numerous sites [14]. However detailed information such as disease types is not disclosed nor there are segmentation labels. Nevertheless, this large real clinical dataset can be used to train the baseline dense motion model.

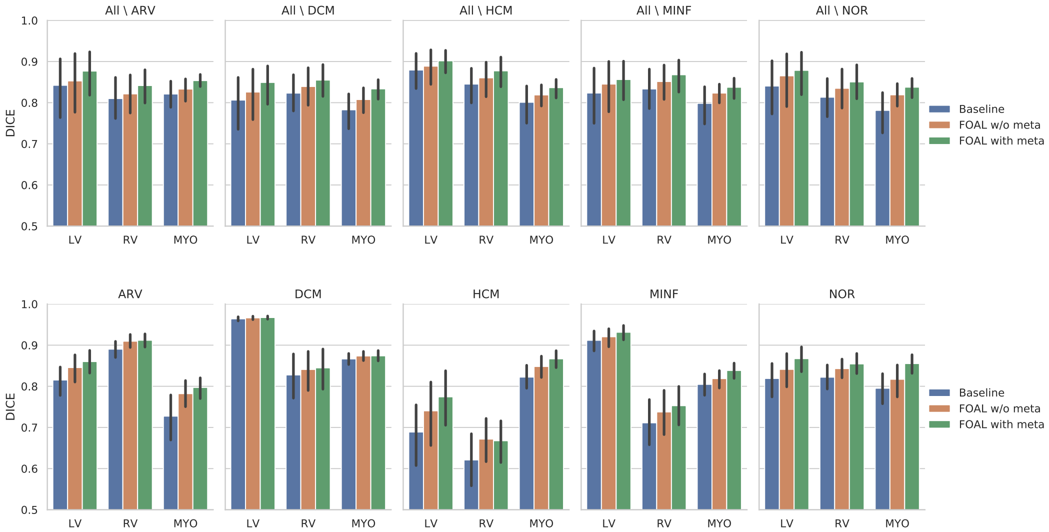

Figure 4. The bar-plots of inside distribution v.s. outside distribution Dice coefficient results for the baseline model, proposed FOAL without meta-learning (FOAL w/o meta) and FOAL with meta-learning (FOAL with meta) for all five folds. Different diseases as outside distributions are presented in different columns. The top row is the inside distribution test and the bottom row is the outside distribution test. The data of the outside distribution disease were excluded in the baseline training and meta-training. Averaged values and standard deviations are presented.

4.2. Implementation Details

For image preprocessing, we normalized the gray value to 0-255 and we applied center cropping and zero padding to adjust image size to 192 × 192. All models are trained and tested on a Tesla V100 workstation. The other implementation details are presented as following.

Dense motion tracker: As for the baseline model, we adopted a lightweight (shallower and narrower) version of the motion prediction network proposed by Qin *et al.* [26]. We halved the number of feature maps of each layer and the number of layers. We set $\alpha_s = 5 \times 10^{-5}$ and $\beta_c = 10^{-6}$ in Eq. (4). The batch size is 20 images. We utilized Adam optimizer with an initial learning rate 10^{-3}.

Online optimizer: The number of update steps $m = 3$ and the number of sampled pairs $K = 24$ in Algorithm 1. We used Adam optimizer with learning rate $\alpha = 10^{-4}$.

Meta learner: We used the number of sampled videos $n = 2$, the number of update steps $m = 5$, and the number of sampled pairs $K = 24$ in the online optimization in Algorithm 2. SGD optimizer is used for online optimizer with a fixed learning rate $\alpha = 10^{-5}$. Adam optimizer is used for the meta-learner with an initial learning rate $\beta = 10^{-5}$ in Algorithm 2. The meta training steps are 6,000.

4.3. Experiment Setups

Inside distribution vs Outside distribution: In data-driven machine learning, we always hypothesize that training samples and testing samples are drawn from the same distribution (inside distribution). The violation of the hypothesis (outside distribution in the testing set) usually gives poor model generalization on the testing set. In this study, we performed five-fold cross-validations in light of the leaving-one-disease-out method on the ACDC dataset. The idea behind this is to separate inside distribution (P_{in}) and outside distribution (P_{out}) in terms of known diseases. Due to the significant cardiac anatomy and dynamic differences between different diseases, one disease category could be viewed as an outside distribution compared to the other 4 diseases. For subjects in the inside distribution set, we separate them into train set ($80 \times 80\% = 64$ subjects) as $p(D_a)$ and $p(D_{meta})$, and test set ($80 \times 20\% = 16$ subjects) as $p(D_{t_{inside}})$. 100% subjects in the outside distribution (20 subjects) set were used in the test set as $p(D_{t_{outside}})$. In this experiment, we trained and evaluated all three compared methods on the ACDC dataset.

Fine-tuning and Generalization: We observed that the proposed FOAL with meta-learning needs to train the meta-learner from a baseline model. In the dense tracking context, it is difficult to train the meta-learner from scratch. However, our idea behind the FOAL is to enable any dense tracker to boost their performance via online optimization through meta-learning. To validate the generalizability, we utilized the Kaggle dataset that is without any meta information. Specifically, we used the 30% subjects of the entire Kaggle dataset as $p(D_a)$ to train the baseline model. We then performed leave-one-disease-out cross-validation on the ACDC dataset. Note that the Kaggle data are only used for training the baseline model while $p(D_{meta})$, $p(D_{t_{inside}})$ and $p(D_{t_{outside}})$ are all from ACDC with the same split in the first experiment. In addition to the leave-one-disease-out cross-validation, starting from the baseline model trained on Kaggle, we also compared a vanilla fine-tuning model to FOAL with meta-learning using 20% of the entire ACDC dataset (100 × 20% = 20 subjects). 100% or 10% of the rest ACDC data were used to train the two models. All 5 categories were mixed.

The vanilla fine-tuning model used the same training parameters as the baseline model except that we changed the learning rate to 10^{-5} to prevent large parameter drift [19].

4.4. Quantitative Metrics

We used the DICE coefficient (Eq. (8)) and Hausdorff distance error (Eq. (9)) as quantitative metrics to evaluate the compared tracking methods on segmentation masks. The metrics are defined as:

$$DICE = \frac{2 \times |S_A \cap S_B|}{|S_A| + |S_B|}, \tag{8}$$

where S_A and S_B are the segmentation mask A and the segmentation mask B, respectively.

$$H(C_A, C_B) = \max_{a \in C_A} \{ \min_{b \in C_B} ||a - b||_2 \}, \tag{9}$$

where a and b are the points on the contour A and the contour B, respectively. $|| \cdot ||_2$ is the Euclidean distance.

5. Results and Discussion

Inside distribution vs outside distribution on ACDC data: The five-fold cross-validation experiment in this part is described in Section 4.3. Fig. 4 depicts all three compared methods (baseline model, FOAL without meta-learning and FOAL with meta-learning) in every cross-validation with test samples drawn from inside or outside distribution. Table 1 and Table 2 summarize Dice and Hausdorff distance results, respectively, for both inside and outside distributions averaged over the five folds. Fig. 4, Table 1 and Table 2 show that the proposed FOAL with meta-learning approach outperforms the baseline tracker. For the inside distribution test, our FOAL with meta-learning increased the Dice by 3.7% and reduced Hausdorff distance error by 1.0 mm on average. It is worth pointing out that even the training and testing are within the same disease distribution, the variations from patients, scanner types, scanner settings, etc. are still large, which can explain the reduced errors from our method compared to the baseline. The largest accuracy improvement occurs on MYO with 4.3% on Dice for both inside distribution and outside distribution. On the zero-shot (outside distribution) dataset, our FOAL with meta-learning achieves superior performance (e.g. on average 3.8% increase on Dice) compared to the baseline. Besides, we observed that FOAL with meta-learning outperforms FOAL without meta-learning consistently. This demonstrates the effectiveness of meta-learning to enhance the adaptation capability of the online optimizer. This result is not surprising because the online optimizer learns how to adapt to a new video using offline meta training on a large number of videos. This capability teaches the online optimizer to find a sub-optimal path to a better solution than the optimizer without meta-learning can.

Table 3. Finetuning experiment with Kaggle baseline training and ACDC inside and outside distribution test sets. Dice coefficients are averaged over the five-fold cross-validation for baseline model trained on Kaggle data (Baseline), fine-tuned model on the ACDC dataset (Finetune) and FOAL with meta-learning (FOAL + meta) on the ACDC dataset. Numbers are shown in mean(std).

Method	LV	RV	MYO
	Inside Distribution Test Set		
Baseline	0.864(0.019)	0.847(0.013)	0.830(0.010)
Finetune	0.861(0.023)	0.850(0.012)	0.827(0.014)
FOAL + meta	**0.880(0.017)**	**0.866(0.010)**	**0.847(0.009)**
	Outside Distribution Test Set		
Baseline	0.874(0.070)	0.796(0.093)	0.841(0.024)
Finetune	0.870(0.070)	0.792(0.094)	0.833(0.031)
FOAL + meta	**0.885(0.059)**	**0.804(0.091)**	**0.849(0.023)**

Fig. 5 depicts the warped segmentation results using corresponding deformation fields which were generated by the baseline model and FOAL with meta-learning. In Fig. 5, ED and ES frames in the video are also illustrated. We observed a significant appearance and shape difference inside the heart region. Referring to annotations, our method improved LV (blue color) and MYO (green color) comparing to the baseline method. Note that the result can not be compared directly with the results in supervised segmentation [26] since our task is unsupervised motion tracking.

Table 4. Finetuning experiment with Kaggle baseline training and 100% and 10% ACDC training dataset. Dice coefficients (mean(std)) for baseline model trained on Kaggle data (Baseline), vanilla fine-tuned model on the ACDC (Finetune) and FOAL with meta-learning on the ACDC (FOAL + meta) are reported.

Method	LV	RV	MYO
	100% of ACDC training data		
Baseline	0.865(0.103)	0.845(0.080)	0.829(0.065)
Finetune	0.865(0.104)	0.854(0.079)	0.831(0.063)
FOAL	**0.881(0.086)**	**0.865(0.070)**	**0.845(0.051)**
	10% of ACDC training data		
Baseline	0.865(0.103)	0.845(0.080)	0.829(0.065)
Finetune	0.864(0.104)	0.845(0.082)	0.824(0.073)
FOAL +meta	**0.882(0.086)**	**0.863(0.071)**	**0.845(0.051)**

Fine-tuning and Generalization: The experiment setup in this part is discussed in Section 4.3. We compared the baseline model trained on Kaggle data (Baseline), a model fine-tuned on ACDC data from the baseline model (Fine-

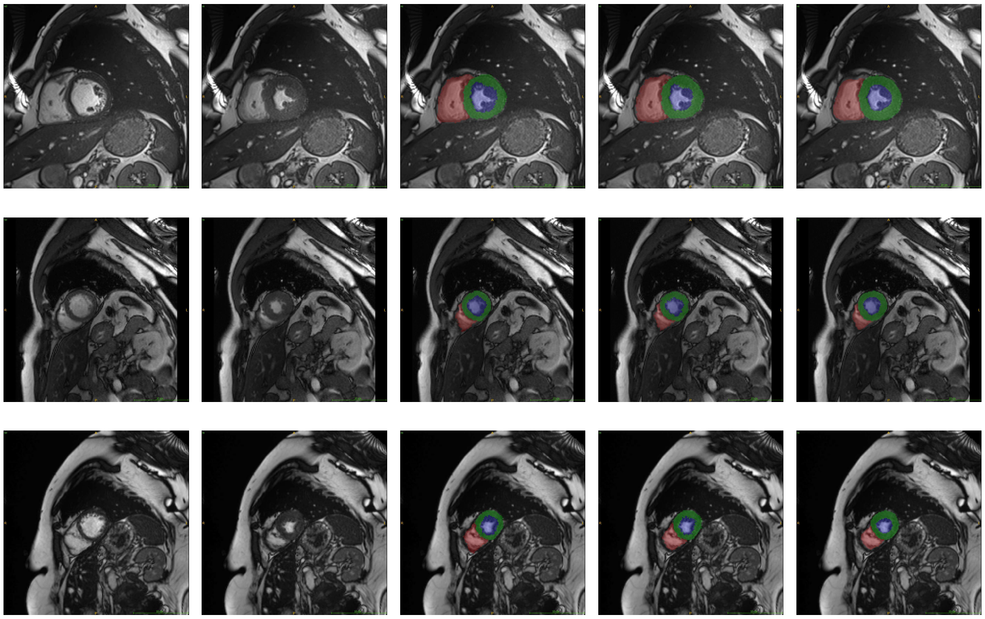

Figure 5. Examples of the tracking results of the mask overlay warped from ED heart phase to ES heart phase. The warp operation utilized deformation fields which were generated from the compared methods. From left to right: the starting frame (ED phase), the final frame (ES phase), baseline model, FOAL with meta-learning and the expert mask annotations. Note that the red mask represents RV, green represents MYO and blue represents LV.

tune) and our proposed FOAL with meta-learning from the baseline model (FOAL+meta). Averaged Dice coefficients among five folds for both inside distribution and outside distribution can be found in Table 3. The baseline model performs comparably well on both distributions except RV. This might be because the Kaggle dataset consists of a variety of cardiac diseases and it has distribution overlaps with both the inside distribution and the outside distribution datasets but not for RV. Fine-tuning the model on the ACDC dataset does not improve the performance. Comparing to the baseline model, our method improved 2.7% on the inside distribution test and 2.4% on the outside distribution test in terms of Dice. Though we test the generalization of the method on CMR datasets, FOAL may have the potential of generalization to other motion(flow) estimation datasets like the KITTI and Sintel Final.

Table 4 shows Dice results for vanilla fine-tuning model and our FOAL with meta-learning using 10% or 100% ACDC training samples. In contrast to the leave-one-disease-out experiments, we did not isolate any disease in the training samples in this experiment and the models were tested on the entire ACDC test set. Vanilla fine-tuning model made the performance slightly worse in the 10% experiment while it slightly improved the accuracy in the 100% experiment comparing to the baseline model. Meanwhile, FOAL with meta-learning gave 1.68% and 1.71% Dice increases on average for both 10% and 100% experiments, respectively. This result is consistent with the above fivefold cross-validation test. In addition, Fig. 4 demonstrates that our FOAL performs comparably well using a small amount of data when it is meta-trained from a strong baseline model.

Our FOAL online optimization algorithm requires 413 ± 8 milliseconds (mean±standard deviation), which we find it completely durable for most current clinical applications.

6. Conclusion

In this work, we proposed a novel online adaptive learning method to minimize the domain mismatch problem in the context of dense cardiac motion estimation. The online adaptor is a gradient descent based optimizer which itself is also optimized by a meta-learner. The meta-learning strategy allows the online optimizer to perform a fast adaption using a limited number of model updates and a small number of image pairs from a single video. The tracking performance is significantly improved in all the zero-shot (outside distribution comparing to the training samples) experimental setups. Also, it is observed that the online adaptor can minimize the tracking errors in the inside distribution tests. Experimental results demonstrate that our methods obtain superior performance compared to the model without online adaption. The pilot study shows the feasibility of applying the method in the context of unsupervised dense motion tracking or deformable image registration. The proposed method provides a practical and elegant approach to an often overlooked problem in existing art. We hope to inspire more discussions and work to benefit other clinical applications suffering from similar issues.

References

[1] Olivier Bernard, Alain Lalande, Clement Zotti, Frederick Cervenansky, Xin Yang, Pheng-Ann Heng, Irem Cetin, Karim Lekadir, Oscar Camara, Miguel Angel Gonzalez Ballester, et al. Deep learning techniques for automatic MRI cardiac multi-structures segmentation and diagnosis: Is the problem solved? *IEEE transactions on medical imaging*, 37(11):2514–2525, 2018. 5

[2] Jingchun Cheng, Yi-Hsuan Tsai, Shengjin Wang, and Ming-Hsuan Yang. Segflow: Joint learning for video object segmentation and optical flow. In *Proceedings of the IEEE international conference on computer vision*, pages 686–695, 2017. 2

[3] Sumit Chopra, Raia Hadsell, Yann LeCun, et al. Learning a similarity metric discriminatively, with application to face verification. In *CVPR (1)*, pages 539–546, 2005. 3

[4] Mathieu De Craene, Gemma Piella, Oscar Camara, Nicolas Duchateau, Etelvino Silva, Adelina Doltra, Jan D'hooge, Josep Brugada, Marta Sitges, and Alejandro F Frangi. Temporal diffeomorphic free-form deformation: Application to motion and strain estimation from 3D echocardiography. *Medical image analysis*, 16(2):427–450, 2012. 2, 3

[5] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015. 2

[6] James P Earls, Vincent B Ho, Thomas K Foo, Ernesto Castillo, and Scott D Flamm. Cardiac MRI: recent progress and continued challenges. *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 16(2):111–127, 2002. 2

[7] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org, 2017. 3, 4, 5

[8] Chelsea Finn, Tianhe Yu, Tianhao Zhang, Pieter Abbeel, and Sergey Levine. One-shot visual imitation learning via meta-learning. *arXiv preprint arXiv:1709.04905*, 2017. 3

[9] Victor Garcia and Joan Bruna. Few-shot learning with graph neural networks. *arXiv preprint arXiv:1711.04043*, 2017. 3

[10] Qiaoying Huang, Dong Yang, Hui Qu, Jingru Yi, Pengxiang Wu, and Dimitris N Metaxas. Dynamic MRI reconstruction with motion-guided network. 2018. 1

[11] Junhwa Hur and Stefan Roth. Iterative residual refinement for joint optical flow and occlusion estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1

[12] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017. 1, 2

[13] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015. 3

[14] kaggle. Data science bowl cardiac challenge data, 2014. Second Annual Data Science Bowl from kaggle, https://www.kaggle.com/c/second-annual-data-science-bowl/data. 5

[15] Julian Krebs, Hervé e Delingette, Boris Mailhé, Nicholas Ayache, and Tommaso Mansi. Learning a probabilistic model for diffeomorphic registration. *IEEE transactions on medical imaging*, 2019. 1, 2, 3

[16] Hsueh-Ying Lai, Yi-Hsuan Tsai, and Wei-Chen Chiu. Bridging stereo matching and optical flow via spatiotemporal correspondence. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1

[17] Matthew Chung Hai Lee, Kersten Petersen, Nick Pawlowski, Ben Glocker, and Michiel Schaap. Tetris: Template transformer networks for image segmentation with shape priors. *IEEE transactions on medical imaging*, 2019. 1

[18] Xin Lei, Liangyu He, Yixuan Tan, Ken Xingze Wang, Xinggang Wang, Yihan Du, Shanhui Fan, and Zongfu Yu. Direct object recognition without line-of-sight using optical coherence. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1

[19] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017. 7

[20] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*, 2018. 3

[21] Pengpeng Liu, Michael Lyu, Irwin King, and Jia Xu. Selflow: Self-supervised learning of optical flow. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1

[22] Simon Meister, Junhwa Hur, and Stefan Roth. Unflow: Unsupervised learning of optical flow with a bidirectional census loss. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 1, 2, 3

[23] Manuel A Morales, David Izquierdo-Garcia, Iman Aganj, Jayashree Kalpathy-Cramer, Bruce R Rosen, and Ciprian Catana. Implementation and validation of a three-dimensional cardiac motion estimation network. *Radiology: Artificial Intelligence*, 1(4):e180080, 2019. 1

[24] Asif Padiyath, Paul Gribben, Joseph R Abraham, Ling Li, Sheela Rangamani, Andreas Schuster, David A Danford, Gianni Pedrizzetti, and Shelby Kutty. Echocardiography and cardiac magnetic resonance-based feature tracking in the assessment of myocardial mechanics in tetralogy of fallot: an intermodality comparison. *Echocardiography*, 30(2):203–210, 2013. 1

[25] Esther Puyol-Antón, Bram Ruijsink, Wenjia Bai, Hélène Langet, Mathieu De Craene, Julia A Schnabel, Paolo Piro, Andrew P King, and Matthew Sinclair. Fully automated myocardial strain estimation from cine MRI using convolutional neural networks. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 1139–1143. IEEE, 2018. 2, 3

[26] Chen Qin, Wenjia Bai, Jo Schlemper, Steffen E Petersen, Stefan K Piechnik, Stefan Neubauer, and Daniel Rueckert. Joint learning of motion estimation and segmentation for cardiac MR image sequences. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 472–480. Springer, 2018. 1, 3, 6, 7

[27] Anurag Ranjan, Varun Jampani, Lukas Balles, Kihwan Kim, Deqing Sun, Jonas Wulff, and Michael J. Black. Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1

[28] Martin Reindl, Christina Tiller, Magdalena Holzknecht, Ivan Lechner, Alexander Beck, David Plappert, Michelle Gorzala, Mathias Pamminger, Agnes Mayr, Gert Klug, et al. Prognostic implications of global longitudinal strain by feature-tracking cardiac magnetic resonance in st-elevation myocardial infarction. *Circulation: Cardiovascular Imaging*, 12(11):e009404, 2019. 1

[29] Daniel Rueckert, Luke I Sonoda, Carmel Hayes, Derek LG Hill, Martin O Leach, and David J Hawkes. Nonrigid registration using free-form deformations: application to breast MR images. *IEEE transactions on medical imaging*, 18(8):712–721, 1999. 2, 3

[30] Jürgen Schmidhuber. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook*. PhD thesis, Technische Universität München, 1987. 3

[31] Gavin Seegoolam, Jo Schlemper, Chen Qin, Anthony Price, Jo Hajnal, and Daniel Rueckert. Exploiting motion for deep learning reconstruction of extremely-undersampled dynamic MRI. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 704–712. Springer, 2019. 1

[32] Dinggang Shen, Hari Sundar, Zhong Xue, Yong Fan, and Harold Litt. Consistent estimation of cardiac motions by 4D image registration. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 902–910. Springer, 2005. 2, 3

[33] Wenzhe Shi, Xiahai Zhuang, Haiyan Wang, Simon Duckett, Duy VN Luong, Catalina Tobon-Gomez, KaiPin Tung, Philip J Edwards, Kawal S Rhode, Reza S Razavi, et al. A comprehensive cardiac motion estimation framework using both untagged and 3-D tagged MR images based on nonrigid registration. *IEEE transactions on medical imaging*, 31(6):1263–1275, 2012. 2, 3

[34] Nicholas B Spath, Miquel Gomez, Russell J Everett, Scott Semple, Calvin WL Chin, Audrey C White, Alan G Japp, David E Newby, and Marc R Dweck. Global longitudinal strain analysis using cardiac MRI in aortic stenosis: Comparison with left ventricular remodeling, myocardial fibrosis, and 2-year clinical outcomes. *Radiology: Cardiothoracic Imaging*, 1(4):e190027, 2019. 1

[35] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: CNNs for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8934–8943, 2018. 1, 2

[36] Qianru Sun, Xinzhe Li, Yaoyao Liu, Shibao Zheng, Tat-Seng Chua, and Bernt Schiele. Learning to self-train for semi-supervised few-shot classification. *arXiv preprint arXiv:1906.00562*, 2019. 3

[37] Catalina Tobon-Gomez, Mathieu De Craene, Kristin Mcleod, Lennart Tautz, Wenzhe Shi, Anja Hennemuth, Adityo Prakosa, Hengui Wang, Gerry Carr-White, Stam Kapetanakis, et al. Benchmarking framework for myocardial tracking and deformation algorithms: An open access database. *Medical image analysis*, 17(6):632–648, 2013. 2, 3

[38] Gabriele Valvano, Agisilaos Chartsias, Andrea Leo, and Sotirios A Tsaftaris. Temporal consistency objectives regularize the learning of disentangled representations. In *Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data*, pages 11–19. Springer, 2019. 1

[39] Davis M Vigneault, Weidi Xie, David A Bluemke, and J Alison Noble. Feature tracking cardiac magnetic resonance via deep learning and spline optimization. In *International Conference on Functional Imaging and Modeling of the Heart*, pages 183–194. Springer, 2017. 3

[40] Liang Wang, Patrick Clarysse, Zhengjun Liu, Bin Gao, Wanyu Liu, Pierre Croisille, and Philippe Delachartre. A gradient-based optical-flow cardiac motion estimation method for cine and tagged MR images. *Medical image analysis*, 57:136–148, 2019. 2

[41] Yang Wang, Peng Wang, Zhenheng Yang, Chenxu Luo, Yi Yang, and Wei Xu. Unos: Unified unsupervised optical-flow and stereo-depth estimation by watching videos. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1

[42] Yunchao Wei, Huaxin Xiao, Honghui Shi, Zequn Jie, Jiashi Feng, and Thomas S Huang. Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7268–7277, 2018. 1

[43] Yan Wu, Mihaela Rosca, and Timothy Lillicrap. Deep compressed sensing. *arXiv preprint arXiv:1905.06723*, 2019. 3

[44] R Xia, T Zhu, Y Zhang, YS Chen, L Wang, JC Liao, YM Li, FJ Lü, and FB Gao. Tracking early reperfused myocardial infarction using cardiac MR. *Sichuan da xue xue bao. Yi xue ban= Journal of Sichuan University. Medical science edition*, 50(4):489–493, 2019. 1

[45] Fan Yang, Yan Zhang, Pinggui Lei, Lihui Wang, Yuehong Miao, Hong Xie, and Zhu Zeng. A deep learning segmentation approach in free-breathing real-time cardiac magnetic resonance imaging. *BioMed research international*, 2019, 2019. 1

[46] Zhichao Yin and Jianping Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1983–1992, 2018. 1

[47] Hanchao Yu, Yang Fu, Haichao Yu, Yunchao Wei, Xinchao Wang, Jianbo Jiao, Matthew Bramlet, Thenkurussi Kesavadas, Honghui Shi, Zhangyang Wang, et al. A novel

framework for 3D-2D vertebra matching. In *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 121–126. IEEE, 2019. 2

[48] Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3712–3722, 2018. 3

[49] Mingliang Zhai, Xuezhi Xiang, Rongfang Zhang, Ning Lv, and Abdulmotaleb El Saddik. Optical flow estimation using dual self-attention pyramid networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019. 1

[50] Qiao Zheng, Hervé Delingette, and Nicholas Ayache. Explainable cardiac pathology classification on cine MRI with motion characterization by semi-supervised learning of apparent flow. *Medical image analysis*, 2019. 1, 3

[51] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. Unsupervised event-based learning of optical flow, depth, and egomotion. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1