Layered X-ray Motion Estimation using Primal-Dual Optimization

Ehsan Mehmood[1], Peter Fischer[1,2], Thomas Pohl[2], Tim Horz[2], Andreas Maier[1]

[1]Pattern Recognition Lab, FAU Erlangen-Nürnberg, Erlangen, Germany
[2]Siemens Healthineers, Forchheim, Germany
esn.mehmood@gmail.com

Abstract. Layered motion estimation (LME) in X-ray fluoroscopy is a challenging, ill-posed and non-convex problem due to transparency effects and the way the image is defined. Minimizing an energy formulation of layered motion estimation is computationally expensive. For clinical usability of this approach, we propose to use primal-dual optimization parallelized using a graphical processing unit (GPU) to reduce the overall run-time of this algorithm.
Experimentally this method is able to substantially reduce target registration error by 70% on manually annotated landmarks on five distinct image sequences compared to the static baseline, similar to prior work on this domain. However, the overall runtime of our method on a conventional GPU is less than 3.3 seconds compared to several minutes for the state of the art. Considering typical frame-rates of X-ray fluoroscopy devices, this runtime makes the application of layered motion estimation feasible for many clinical workflows.

1 Introduction

X-ray fluoroscopy, due to its very good spatial and temporal resolution, is an important modality for clearly visualizing human body functions and internal structures, and is commonly used for guidance in minimally invasive interventions. Motion estimation is useful for many clinical applications in X-ray fluoroscopy such as blood flow monitoring, detection of dead tissues and tracking of kidney stones and tumors. Cardiac and respiratory motion can be compensated to improve visibility and perceptibility in Coronary DSA [1]. Fusion of previously acquired roadmaps requires a motion estimate to accurately display overlays on live fluoroscopic images [2].

Since X-ray images are formed by transparent projections of a 3-D volume onto a 2-D plane, there is information loss as well as transparency effects in the images. To solve the transparency problem in X-ray image registration, motion layers are introduced with a goal to calculate a 2-D motion field for each layer. This approach involves estimation of both layers and motions with information on neither available initially. Their dependency on each other causes a chicken-and-egg problem since computation of layers requires motion estimated and vice versa. Szeliski et al. assume parametric motion to simplify the problem and describe two methods to compute layers, using max and min composites for sequential initialization and using constrained least squares optimization for iterative refinement [3]. Preston et al. introduce a method to jointly estimate layers and motions by using a total-variation layer gradient penalty and a smoothness prior for motions [4]. To generate physiologically plausible motions, Fischer et al. proposed to use surrogate signals to define a model of the layer motions [5] and an alternating minimization scheme to calculate motions and layers.

The existing work by Fischer et al. [5] is able to plausibly estimate layers and motions. However, the runtime is several minutes and thus not feasible for time-critical clinical applications. We propose to solve the layered motion estimation problem using primal-dual optimization. Compared to other optimizers, primal-dual methods are simple, easily parallelizeable, and can handle non-smooth problems naturally. They involve splitting of the main problem into simpler sub-problems that can be solved efficiently using computationally efficient proximity operators. Thus, we implement layered motion estimation on the graphical processing unit (GPU) to achieve a clinically acceptable time. In the experiments, we demonstrate that a similar accuracy of motion estimation as in [5] is achieved. Additionally, we analyze and compare the runtimes of the different algorithms.

2 Methods

2.1 Layered Motion Estimation

We are interested in separating transparent X-ray images, denoted in this paper as $\boldsymbol{i}$ (with dimensions $N_t \times H \times W$) into multiple layers $\boldsymbol{l}_n$ (with dimensions $N_l \times H \times W$) and motions $\boldsymbol{v}_n$ (with dimensions $N_l \times H \times W \times 2$) that each layer n undergoes, where N_t, N_l, W, H are number of images in the sequence, number of layers, width of an image and height of an image respectively. In mathematical terms, we can define our image as

$$I = \mathbf{a}\left(\boldsymbol{l}, \boldsymbol{v}\right) + \boldsymbol{\eta} \tag{1}$$

where $\mathbf{a}\left(\boldsymbol{l}, \boldsymbol{v}\right)$ is a function that creates an image sequence estimate using bilinear remapping of layers and motions and $\boldsymbol{\eta}$ is introduced to account for model errors and observation noise in the log-transformed X-ray model [4].

Moreover, to make motions physiologically plausible, calculated base motions $\boldsymbol{\nu}_n$ are scaled using surrogate signals to obtain motions according to [5]:

$$\boldsymbol{v}_n\left(\boldsymbol{x}, t\right) = s_n\left(t\right) \cdot \boldsymbol{\nu}_n\left(\boldsymbol{x}\right) \tag{2}$$

In discretized form, the energy equation of the layered motion estimation (LME) problem can be written as

$$\min_{\boldsymbol{l},\boldsymbol{v}} \underbrace{\left\|\mathbf{a}\left(\boldsymbol{l}, \boldsymbol{v}\right) - I\right\|_1}_{\text{Data term}} + \underbrace{\lambda_l \left\|\nabla \boldsymbol{l}\right\|_{2,1}}_{\text{Layer regularizer}} + \underbrace{\lambda_v \left\|\nabla \boldsymbol{v}\right\|_{2,1}}_{\text{Motion regularizer}} \tag{3}$$

where ∇ is the gradient operator and λ_l and λ_v are the regularizer weights. The layer and motion regularization terms in this setup are to ensure layer and motion smoothness. As proposed by Fischer et al. we solve this energy minimization problem through alternate minimization by keeping one variable (layers or motions) constant and minimizing energy function with respect to the other [5].

2.2 Primal-Dual Minimization

Primal-dual minimization minimizes a given problem with respect to its primal and dual form. For the given general problem, its primal form

$$\boldsymbol{x}^* \in \underset{\boldsymbol{x}}{\operatorname{argmin}}\, G(\boldsymbol{x}) + H(\boldsymbol{L}\boldsymbol{x}) \tag{4}$$

and dual form

$$\boldsymbol{y}^* \in \underset{\boldsymbol{y}}{\operatorname{argmin}}\, G^*(-\boldsymbol{L}^T\boldsymbol{y}) + H^*(\boldsymbol{y}) \tag{5}$$

can be solved using Chambolle-Pock algorithm as proposed by Chambolle et al. [6], where $\boldsymbol{x} \in \mathbb{R}^N$ and $\boldsymbol{y} \in \mathbb{R}^K$ are vectors (primal and dual solutions) in real Hilbert spaces, G and H are proper, convex and simple functions, and $\boldsymbol{L} : \mathbb{R}^N \to \mathbb{R}^K$ is a bounded linear operator.

Layer Minimization For layer minimization we can develop an algorithm using appropriate proximity operators for our functions similar to Sidky et al. [7]. Algorithm 1 shows layer minimization algorithm which involves solving with respect to three variables, the primal variable $\boldsymbol{l}$ and the dual variables $\boldsymbol{p}$ and $\boldsymbol{q}$. $\boldsymbol{A}_l$ is a matrix encoding the application of the function $\boldsymbol{a}(\boldsymbol{l}, \boldsymbol{v})$ from Eq. (3) to $\boldsymbol{l}$, which is linear in $\boldsymbol{l}$ assuming constant motion.

Algorithmus 1 : Pseudo-code for N-steps $l_1 - \text{TV}$ Chambolle-Pock algorithm for layers. The constant K is the $l_2 - \text{norm}$ of the matrix defining $\boldsymbol{L}$, τ and σ are non-negative step-size parameters, θ is the update coefficient for primal variable, j is the iteration index and J are the total predefined number of iterations.

$K \leftarrow \|(\mathbf{A}_l, \nabla)\|_2 ; \tau \leftarrow 1/K; \sigma \leftarrow 1/K; n \leftarrow 0; \theta \leftarrow 1; j \leftarrow 0$
$\boldsymbol{l}_0 \leftarrow 0, \bar{\boldsymbol{l}}_0 \leftarrow \boldsymbol{l}_0, \boldsymbol{p}_0 \leftarrow 0, \boldsymbol{q}_0 \leftarrow 0$
while $j \leq J$ **do**
  $\boldsymbol{p}_{j+1} \leftarrow \left(\boldsymbol{p}_j + \sigma\left(\mathbf{A}_l\bar{\boldsymbol{l}}_j - \boldsymbol{i}\right)\right) / \max\left(\mathbf{1}_D, \left|\boldsymbol{p}_j + \sigma\left(\mathbf{A}_l\bar{\boldsymbol{l}}_n - \boldsymbol{i}\right)\right|\right)$
  $\boldsymbol{q}_{j+1} \leftarrow \lambda\left(\boldsymbol{q}_j + \sigma\nabla\bar{\boldsymbol{l}}_j\right) / \max\left(\lambda\mathbf{1}_D, \left|\boldsymbol{q}_j + \sigma\left(\nabla\bar{\boldsymbol{l}}_j\right)\right|\right)$
  $\boldsymbol{l}_{j+1} \leftarrow \boldsymbol{l}_j - \tau\mathbf{A}_l^T\boldsymbol{p}_{j+1} - \tau\nabla^T\boldsymbol{q}_{j+1}$
  $\bar{\boldsymbol{l}}_{j+1} \leftarrow \boldsymbol{l}_{j+1} + \theta\left(\boldsymbol{l}_{j+1} - \boldsymbol{l}_j\right)$
  $j \leftarrow j + 1$
end
return $\boldsymbol{l}_J$

Motion Minimization Similarly, Algorithm 1 can be modified to develop a minimization algorithm for motions with change of primal variable, $\boldsymbol{l}_n \rightarrow \boldsymbol{v}_n$ and remapping function $\boldsymbol{A}_l \rightarrow \boldsymbol{A}_v$, where $\boldsymbol{A}_v$ is approximating the application of $\boldsymbol{a}(\boldsymbol{l}, \boldsymbol{v})$ to $\boldsymbol{v}$. $\boldsymbol{A}_v$ is not linear in $\boldsymbol{v}$, but the algorithm still converges empirically.

2.3 Implementation

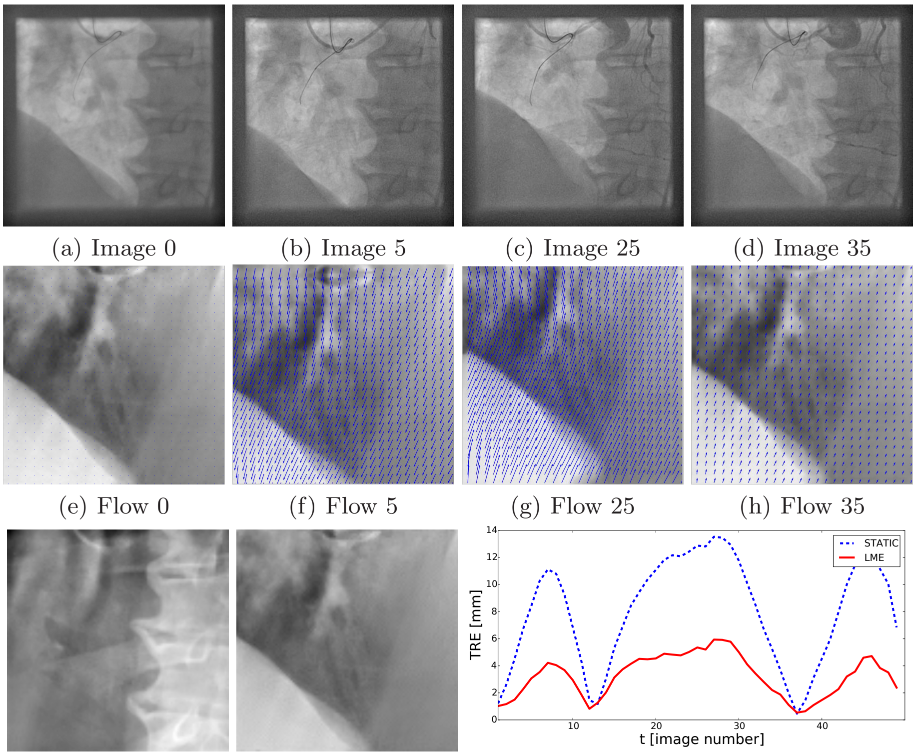
The implementation for the layered motion estimation algorithm is based on a coarse-to-fine method and a two-layer model, a static layer and a respiratory layer. The surrogate signal for the static layer is $s_1(t) = 0$ to describe static components in the images. For the respiratory layer, the surrogate signal $s_2(t)$ is extracted from the intensities of the entire X-ray image sequence using manifold learning. The regularisation term weights for layer and motion smoothness are empirically calculated and set to $\lambda_L = 0.05$ and $\lambda_{\boldsymbol{x}} = 0.025$. Similarly the step sizes for layer and motion minimization problems are also empirically calculated. For all the results obtained during the experiments, an Nvidia Quadro K5000 GPU and an Intel Xeon E5-1650 processor were used. The code was optimized with respect to minimum data transfer between CPU and GPU and data spread over maximum number of threads on the GPU.

3 Experiments and Results

The experiments were calculated on 5 clinical data sets containing 50 frames each at a maximum resolution of 128×128 pixels. As a measure of error, target registration error (TRE) was calculated using differences between each point on a manually annotated curve on landmark regions and the computed curve [5]. To obtain a benchmark for comparison, layered motion estimation was implemented using a L-BFGS-B minimizer with tolerance level of 10^{-6}, similar to [5].

Fig. 1 shows results of the proposed algorithm on an image sequence. Fig. 1(a)-(d) show four frames from the image sequence while Fig. 1(e)-(h) show respective calculate motions on those frames. Fig. 1(i) shows the calculated static layer and Fig. 1(j) the respiratory layer while Fig. 1(k) shows TRE over time in comparison to the dotted line which represents the case when there is no motion in the sequence.

The aim of this work was to bring the overall runtime into a clinically feasible range. Fig. 2(a) shows the improvements in overall runtime between the different implementations of this algorithm, primal-dual (PD) and L-BFGS-B (QN) executed on CPU and GPU. Moreover, the decrease in time should not compromise on the quality of results. To study the performance of our primal-dual algorithm relative to L-BFGS-B, we tested the two versions on 5 distinct image sequences and calculated TRE error means and standard deviations. When compared to TRE on a stationary sequence, 4.98 ± 3.14 mm, L-BFGS-B reduced the TRE to 1.45 ± 0.50 mm and primal-dual also showed similar results, 1.46 ± 0.47 mm for empirically tuned step-sizes with respect to each image sequence.



(a) Image 0 (b) Image 5 (c) Image 25 (d) Image 35

(e) Flow 0 (f) Flow 5 (g) Flow 25 (h) Flow 35

(i) Static layer (j) Respiratory layer (k) TRE over time

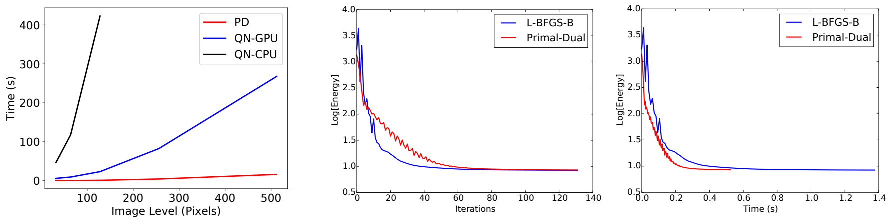
Fig. 1. Layered motion estimation results on an image sequence

To compare the performance of the two methods in more detail, we look at the convergence of both algorithm versions for same data set. Fig. 2(b) shows the energy convergence with respect to number of iterations while Fig. 2(c) shows the convergence of both algorithm versions with respect to time.

4 Conclusion and Outlook

Use of a faster and easily parallelizeable primal-dual method instead of L-BFGS-B minimization together with implementation on the GPU helped us to reduce the overall runtime of the algorithm from 642 seconds to less than 3.3 seconds for an image sequence containing 50 frames while maintaining the quality of results. This algorithm can handle a rate of 15 frames per second which is usually quite high for an x-ray fluoroscopy machine. Moreover, it was able to reduce the overall TRE by 70% for step-size parameters tuned with respect to each image sequence, which means primal-dual method performed on par with L-BFGS-B method.

In future work, more layers can be incorporated into this model especially to cater for cardiac motion. Automatic step-size calculation for primal-dual op-



(a) Time plot (b) Conv. by iterations (c) Conv. by time

Fig. 2. Comparison: Primal-Dual vs. L-BFGS-B

timizer is an important improvement that can be worked on in order to rely less on empirically calculated values. Although this algorithm has been tested on clinical data, it still needs to be incorporated into a clinical prototype and tested on live subjects.

Acknowledgements The authors gratefully acknowledge funding of the Erlangen Graduate School in Advanced Optical Technologies (SAOT), by the German Research Foundation (DFG) in the framework of the German excellence initiative and by Siemens Healthineers. The concepts and information presented in this paper are based on research and are not commercially available.

References

1. Zhu Y, Prummer S, Wang P, et al. Dynamic layer separation for coronary DSA and enhancement in fluoroscopic sequences. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer; 2009. p. 877–884.
2. Brost A, Liao R, Strobel N, Hornegger J. Respiratory motion compensation by model-based catheter tracking during EP procedures. Medical Image Analysis. 2010;14(5):695–706.
3. Szeliski R, Avidan S, Anandan P. Layer extraction from multiple images containing reflections and transparency. In: Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on. vol. 1. IEEE; 2000. p. 246–253.
4. Preston JS, Rottman C, Cheryauka A, et al. Multi-layer deformation estimation for fluoroscopic imaging. In: International Conference on Information Processing in Medical Imaging. Springer; 2013. p. 123–134.
5. Fischer P, Pohl T, Maier A, Hornegger J. Surrogate-Driven Estimation of Respiratory Motion and Layers in X-Ray Fluoroscopy. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer; 2015. p. 282–289.
6. Chambolle A, Pock T. A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of Mathematical Imaging and Vision. 2011;40(1):120–145.
7. Sidky EY, Jakob H, Pan X. Convex optimization problem prototyping for image reconstruction in computed tomography with the Chambolle-Pock algorithm. Physics in Medicine and Biology. 2012;57(10):3065.