PROCEEDINGS OF SPIE

Application of Proximal Alternating Linearized Minimization (PALM) and inertial PALM to dynamic 3D CT

Nargiza Djurabekova, Andrew Goldberg, Andreas Hauptmann, David Hawkes, Guy Long, et al.

Application of Proximal Alternating Linearized Minimization (PALM) and inertial PALM to dynamic 3D CT

Nargiza Djurabekova[1], Andrew Goldberg[1,3], Andreas Hauptmann[1,5], David Hawkes[1], Guy Long[2], Felix Lucka[1,4], and Marta Betcke[1]

[1]University College London
[2]Curvebeam
[3]RNOH NHS Trust
[4]Centrum Wiskunde & Informatica

Keywords: Cone beam CT, dynamic CT, non-convex optimisation, PALM, iPALM, optical flow regularisation, variational methods

1. INTRODUCTION

1.1 Motivation and background

The foot and ankle is a complex structure consisting of 28 bones and 30 joints that changes from being completely mobile when positioning the foot on the floor to a rigid closed pack position during propulsion such as when running or jumping. An understanding of this complex structure has largely been derived from cadaveric studies.[1] In vivo studies have largely relied on skin surface markers and multi-camera systems[2] that are unable to differentiate small motions between the bones of the foot . MRI and CT based studies have struggled to interpret functional weight bearing motion as imaging is largely static and non-load bearing. Arthritic diseases of the foot and ankle are treated either by fusion of the joints to remove motion, or joint replacement to retain motion. Until a better understanding of the biomechanics of these joints can be achieved, it is difficult to determine which treatment offers the best functional outcome (see Figure 1).

Figure 1: Left: Ankle arthroplasty or ankle replacement. Right: Ankle arthrodesis or ankle fusion. *Images courtesy of Mr Andrew Goldberg MD FRCS(Tr & Orth)*

The aim of the current project is to develop reconstruction techniques for a fully 4D (3D + time) Cone beam CT (CBCT) of the foot and ankle using a weight-bearing CBCT machine (Curvebeam, Hatfield, PA, USA) such as PedCat or LineUP.

1.2 CT and motion

Motion arises in many imaging applications whenever the imaging target exhibits dynamics at time scales of the order of image acquisition time. This can lead to blurring and other artifacts which can affect diagnostic validity of the reconstruction from the acquired data. To this end, a lot of research concentrates on motion correction[3] although new application relying on dynamic imaging are constantly emerging, particularly in the field of cardiac imagining.

In line with this development the goal of the present paper is to reconstruct the dynamics of the more rigid foot and ankle structure. This is done by using *optical flow* (or its natural extension into the 3D space - *scene flow*). Optical flow, or the transport term, is the assumption of brightness constancy over all time-frames. That is, we assume that all elements of the foot present in one frame are going to appear in the next frame, with only some fairly small changes in location and orientation. Such changes can be described using a vector field, also known as the motion field, which is reconstructed along the linear attenuation coefficient.

2. METHODOLOGY

2.1 Setup

As it is unrealistic to expect to reconstruct the full 3D foot with just a single projection per time-frame, we assume that the foot can move in a semi-consistent periodic motion. This would allow us to bin several projections to help reconstruct individual time-frames of the full motion. This situation is simulated as follows. A set of bones, extracted from a segmented full data reconstruction of the foot and ankle, are arranged and animated to create the phantom that is used to test the algorithms described in this paper (see Acknowledgements). ASTRA toolbox[4] emulates the forward conebeam projections at quasi-randomly selected angles. The angles were sampled using stratified random sampling, i.e. to get three projection angles, each would be chosen randomly from a designated third of all available angles.

2.2 Algorithms and the problem formulation

We begin by formulating our problem, similarly to 5,6, as a minimization problem where regularization parameters are used to find balance between the data fidelity term, the optical flow term (briefly mentioned in sect. 1) as well as TV regularization on the image u and the motion field $\boldsymbol{v}$. So, the discretized spatio-temporal CT problem can be stated as

$$(u^*, \boldsymbol{v}^*) = \underset{u \geq 0, \boldsymbol{v}}{\arg\min} \Big\{ \sum_t^T \frac{1}{2}||Au_t - f_t||_2^2 + \alpha TV(u_t) + \beta TV_s(\boldsymbol{v}_t) + \frac{\gamma}{2}||\nabla_t u_t + (\nabla u_t) \cdot \boldsymbol{v}_t||_2^2 \Big\}, \quad (1)$$

where A is the forward operator, f is the measurement, the last term is the aforementioned optical flow term and α, β and γ are the regularization parameters. Lastly TV_s is just TV that's applied to each individual component of the motion field $\boldsymbol{v}$

This problem is non-convex and can be re-arranged to satisfy the general function

$$\text{minimize}_{x,y} F(x,y) := f_1(x) + f_2(y) + H(x,y), \quad (2)$$

that can be solved by PALM[7] or iPALM,[8] with f_1 and f_2 being the TV and TV_s respectively and H, the coupling function with Lipschitz continuous gradient, being the sum of the data fidelity and the transport terms. Then the PALM algorithm is described in Alg.1. iPALM, the inertial counterpart of PALM is similar but with some of the momentum saved from the previous iterates. The main way the two algorithms differ is in their convergence, with PALM converging in a strictly decreasing fashion and iPALM usually converging faster, as can be seen in figure 2.

Algorithm 1 PALM

1: **Initialization:** Choose any $u^0 \in \mathbb{R}^2(\times\mathbb{R}) \times \mathbb{N}_t$ and $\boldsymbol{v}_0 \in \mathbb{R}^2(\times\mathbb{R}) \times \mathbb{N}_t \times \mathbb{N}_s$, where $\mathbb{N}_t$ is the time step dimension and $\mathbb{N}_s$ is the spatial component dimension with $s = 1, 2(, 3)$.
2: For every $k = 0, 1, ...$ generate a sequence $\{(u^k, \boldsymbol{v}^k)\}$
3: **Repeat**
4: For some $\gamma_1 > 1$, let $c_k = \gamma_1 L_1(\boldsymbol{v}^k)$

$$u^{k+1} \in \mathrm{prox}_{c_k}^{f_1}\big(u^k - \frac{1}{c_k}\nabla_u H(u^k, \boldsymbol{v}^k)\big) \qquad (3)$$

5: For some $\gamma_2 > 1$, let $d_k = \gamma_2 L_2(u^{k+1})$

$$\boldsymbol{v}^{k+1} \in \mathrm{prox}_{d_k}^{f_2}\big(u^k - \frac{1}{d_k}\nabla_{\boldsymbol{v}} H(u^{k+1}, \boldsymbol{v}^k)\big) \qquad (4)$$

---

3. RESULTS AND DISCUSSION

The algorithms were implemented in MATLAB 2017a, using gpuArrays of resolution 128^3 pixels, with a total of 19 time-frames. The appropriate regularization parameters (from (1)) were found in a systematic manner. First setting $\gamma = 1$, which assumes that the transport term has the same weight as the data fidelity term, ignoring β by setting it to 0 and just finding the right α. Once an α that denoises the image sufficiently well is found, β is adjusted to denoise the motion field. Lastly, γ can be slightly adjusted if needed to improve the overall quality of both outputs.

In order to find the appropriate step lengths, the respective Lipschitz constants are estimated in every iteration for the image u and the motion field $\boldsymbol{v}$ using power iteration. This is computationally quite costly, which is why in practice the Lipschitz constant for the image (which barely changes over the course of all iterations) is estimated once and remains fixed for the subsequent iterations.

We highlight a couple of conclusions that can be drawn from results in figures 3 and 2. Firstly, we note that using 3 projection angles, both PALM and iPALM seem to produce better reconstruction than FISTA (an excellent reconstruction algorithm for static objects). Secondly, iPALM converges already on the 1100th iteration with a tolerance of 10^{-4} , while PALM doesn't reach full convergence even after 3000 iterations. And finally it is interesting to observe that the over-smoothing effect of PALM is likely to dissapear with full convergence, as is the case wuith
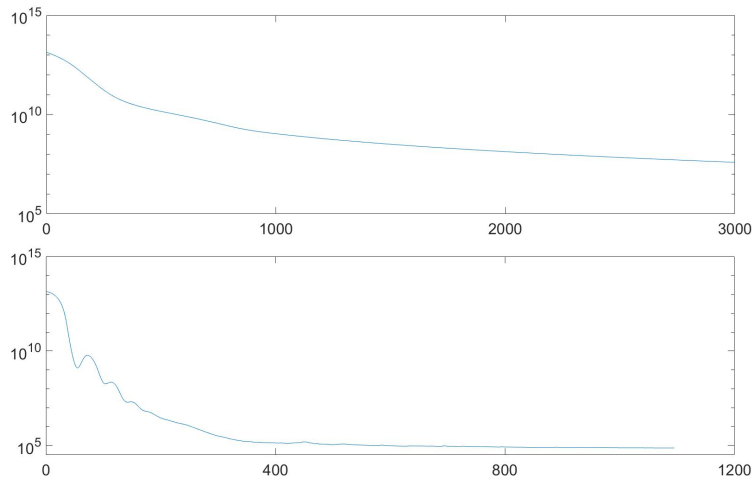
Figure 2: Objective function/energy values plotted over all iterations. (a) for PALM and (b) for iPALM. Note that PALM was stopped due to reaching the maximum iteration of 3000, while iPALM converged at about the 1100th iteration with the tolerance of 10^{-4} between consecutive image iterations.
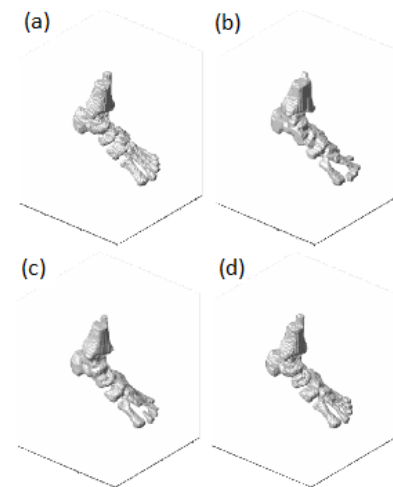

Figure 3: Isosurface plots of the phantom and its reconstructions using 3 projection angles at time-frame 11. (a) Original phantom, (b) FISTA,[9] (c) PALM, (d) iPALM.

iPALM (see figure 3), returning a reconstruction that has similar unevenness as the original phantom.

While even iPALM reconstruction is not perfect and computationally quite heavy, it is encouraging to see that dynamic 3D CT is not completely out of reach. By getting better reconstruction techniques, we are likely to better understand the complex motion in the ankle and foot and design better treatment to help patients.

Acknowledgments

REFERENCES

[1] C. Nester, "Lessons from dynamic cadaver and invasive bone pin studies: do we know how the foot really moves during gait?," *Journal of Foot and Ankle Research* **2**(1), 18 (2009).

[2] J. Leitch, J. Stebbins, and A. Zavatsky, "Subject-specific axes of the ankle joint complex," *Journal of Biomechanics* **43**(15), 2923 – 2928 (2010).

[3] S. Jang, S. Kim, M. Kim, *et al.*, "Head motion correction based on filtered backprojection for x-ray ct imaging," *Medical Physics* **45**(2), 589–604 (2018).

[4] W. van Aarle, W. Palenstijn, J. Cant, *et al.*, "Fast and flexible x-ray tomography using the astra toolbox," *Opt. Express* **24**, 25129–25147 (2016).

[5] M. Burger, H. Dirks, L. Frerking, *et al.*, "A variational reconstruction method for undersampled dynamic x-ray tomography based on physical motion models," *Inverse Problems* **33**(12), 124008 (2017).

[6] M. Burger, H. Dirks, and C.-B. Schonlieb, "A variational model for joint motion estimation and image reconstruction," *SIAM J. Imaging Sciences* **11**(1), 94 – 128 (2018).

[7] J. Bolte, S. Sabach, and M. Teboulle, "Proximal alternating linearized minimization for nonconvex and nonsmooth problems," *Mathematical Programming* **146**, 459–494 (2014).

[8] T. Pock and S. Sabach, "Inertial proximal alternating linearized minimization (ipalm) for nonconvex and nonsmooth problems," **9**, 1756–1787 (2016).

[9] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences* **2**(1), 183–202 (2009).