

Motion Compensated Weighted Filtered Backprojection Considering Rebinning Process

Nora Steinich^a, Johan Sunnegårdh^a, Harald Schöndube^a

^aSiemens Healthineers, Forchheim, Germany

ABSTRACT

Motion during CT scans causes artifacts that can severely degrade the image quality. We propose a motion compensation algorithm that can be combined with reconstruction algorithms that contain a rebinning step like the weighted filtered backprojection algorithm. Therefore, we assume that the motion present during acquisition is known, and we extend the backprojection step in the reconstruction to consider this motion, reducing the motion artifacts in the resulting images. Furthermore, we propose a combination of our motion compensation algorithm with two versions of an iterative weighted filtered backprojection algorithm.

Keywords: Computed Tomography, Motion Compensation, Rebinning, Weighted Filtered Backprojection, Iterative Weighted Filtered Backprojection

1. INTRODUCTION

Motion during image acquisition in a CT scan causes artifacts that can severely degrade the image quality and lower the diagnostic value of the reconstructed images. The motion leads to an inconsistency in the acquired data and hence to artifacts like blurring, streaking or ghost images. The motion is often caused by patient movement, however it can also originate in motion of the scanner itself, for example in mobile scanner systems like a moving or sliding gantry. Since a complete prevention of motion during the scan can be difficult, a solution is required that compensates the present motion and thus improves the quality of the resulting images.

There are many algorithms proposed in the literature that attempt motion compensation in CT reconstruction. One method worth mentioning is the one by Hahn et al. [1], based on partial angle reconstruction. Another interesting motion compensation method is proposed by Bhagalia et al. [2].

In this paper we focus on cases where the motion present during acquisition, or an approximation thereof, is already known. The motion can either be measured with sensors or estimated using a motion estimation method like the one proposed by Bruder et al. [3]. A commonly used method for motion compensation in CT reconstruction with known motion is the method described by Schäfer et al. [4]. Their motion correction algorithm is applied in the reconstruction process during the backprojection step. Every voxel of the voxel volume that should be reconstructed is virtually shifted according to the motion present at the moment of acquisition. They included their method into the reconstruction algorithm of Feldkamp, Davis, and Kress [5] and achieved very good results.

However, there are several popular reconstruction algorithms, like the weighted filtered backprojection (WFBP) algorithm by Stierstorfer et al. [6] which rely on a rebinning step that, when combined with the method of Schäfer et al., will cause artifacts remaining in the final result. In the following we propose an extension of the algorithm by Schäfer et al. that can also handle reconstruction algorithms containing a rebinning method. Furthermore, we propose the combination of the described motion compensation algorithm with an iterative reconstruction method.

2. METHODS

2.1 Motion Compensated Weighted Filtered Backprojection

In this paper we combine the motion compensation method proposed by Schäfer et al. [4] with the WFBP algorithm of Stierstorfer et al. [6], a 3D filtered backprojection algorithm for multislice spiral CT. However, our proposed motion compensation method works with any CT reconstruction algorithm using rebinning and backprojection steps.

In the rebinning step of the WFBP algorithm the projection images acquired in the cone beam geometry are virtually rearranged to form semi-parallel projection images better suited for the backprojection step. The cone beam geometry is described using α as the rotation angle, β as the horizontal opening angle of a ray and q as the cone parameter describing the vertical cone angle of a ray. For the semi-parallel projection images the rotation angle is described as θ and (p, q) denote the detector columns and rows, respectively. The geometry is only semi-parallel, as the rays are only parallel along the detector rows, but not along the columns. In the rebinning step the rays of the initially obtained cone beam projection images $P(\alpha, \beta, q)$ are rearranged to form semi-parallel projection images $P(\theta, p, q)$ using the rebinning formulas

$$\theta = \alpha + \beta \quad p = R \sin(\beta) \quad (1)$$

with R describing the distance from source to isocenter.

The backprojection step in the WFBP algorithm iterates through all rebinned and filtered projection images $P(\theta, p, q)$ and for each projection image it iterates through all voxels \mathbf{x} of the voxel volume to be reconstructed. For each voxel \mathbf{x} , the corresponding position (p, q) in the projection image $P(\theta, p, q)$ the voxel has been projected to during acquisition is calculated. The value at position (p, q) is then added to voxel \mathbf{x} in the voxel volume. This is performed for all voxels and all projection images until the entire voxel volume has been reconstructed.

For our motion compensation method we assume that the motion present during acquisition is known. For every projection image $P(\alpha, \beta, q)$ a motion state \mathbf{M}_α is required. Each motion state \mathbf{M}_α holds the information about the position of the scanned object relative to the scanner at the moment of the acquisition of $P(\alpha, \beta, q)$.

Our proposed algorithm is based on the method by Schäfer et al. [4], extending it to work with reconstruction algorithms containing a rebinning step like the WFBP algorithm. When combining the WFBP algorithm with Schäfer's method the backprojection step of the reconstruction algorithm is extended by moving every voxel \mathbf{x} in the voxel volume to be reconstructed according to the known motion pattern. As described above, the backprojection is performed by iterating for each projection image $P(\theta, p, q)$ through all voxels \mathbf{x} . Knowing the motion state \mathbf{M}_α present during the acquisition of the projection image $P(\alpha, \beta, q)$ obtained at rotation angle α , we can use this information to move each voxel \mathbf{x} accordingly, assuming $\alpha = \theta$. For each projection image $P(\theta, p, q)$ each voxel position \mathbf{x} is recalculated to $\mathbf{x}_{\mathbf{M},\alpha}$ considering the motion state $\mathbf{M}_{\alpha=\theta}$. Using the new position $\mathbf{x}_{\mathbf{M},\alpha}$ the corresponding position (p, q) in the projection image $P(\theta, p, q)$ is calculated and the value of this position is added to the value at voxel position \mathbf{x} . This is then repeated for all voxels in the voxel volume and for all obtained projection images. This leads to a more accurate reconstruction, as the motion present during acquisition is considered. However, when combining the WFBP algorithm with the motion compensation method proposed by Schäfer et al. in this manner, some motion artifacts remain in the final result, as the rebinning process is not considered.

Algorithm 1 Motion Compensated WFBP

```

1: for all projection images  $P(\theta, p, q)$  do
2:   for all voxels  $\mathbf{x}$  of the voxel volume do
3:     1) Calculate new voxel position  $\mathbf{x}_{\mathbf{M},\theta}$  using  $\mathbf{M}_{\alpha=\theta}$ 
4:     2) Calculate  $(p, q)$  using  $\mathbf{x}_{\mathbf{M},\theta}$ 
5:     repeat
6:       3) Calculate  $\mathbf{M}'$  using  $p/p'$ 
7:       4) Calculate  $\mathbf{x}'_{\mathbf{M},\theta}$  using  $\mathbf{M}'$ 
8:       5) Calculate  $(p', q')$  using  $\mathbf{x}'_{\mathbf{M},\theta}$ 
9:     until convergence
10:    6) Add value at position  $(p', q')$  to voxel  $\mathbf{x}$ 
11:   end for
12: end for

```

Due to the rebinning the assumption that for every projection image $P(\theta, p, q)$ one motion state \mathbf{M}_α needs to be considered during backprojection is not accurate. This assumption only holds for the projection images $P(\alpha, \beta, q)$. The rebinning process however rearranges the rays such that the rebinned projection images each consist of information obtained at different time points and hence depicting different motion states. The available motion state \mathbf{M}_α is no longer correct for the entire projection image $P(\theta, p, q)$, but only for the column where $p = 0$, as only there $\theta = \alpha$ holds. Using our

geometry, the columns right of the column $p = 0$ contain previous motion states and columns towards the left contain subsequent motion states. Hence using a motion state \mathbf{M}_α for an entire projection image $P(\theta, p, q)$ in the backprojection will lead to motion artifacts remaining in the final result.

We propose a motion compensation method as described in algorithm 1 extending Schäfer's method to consider the rebinning process, enabling a combination with the WFBP reconstruction algorithm. Steps 1) and 2) are the steps as described in the method of Schäfer et al., while steps 3)-6) are our proposed extension. Our method uses the motion state $\mathbf{M}_{\alpha=\theta}$ to calculate a new position $\mathbf{x}_{\mathbf{M},\theta}$ of the voxel \mathbf{x} , which is then used to calculate an initial position (p, q) in the projection image $P(\theta, p, q)$ just as described by Schäfer et al. This position (p, q) is then used to choose a more accurate motion state \mathbf{M}' . Using the column position p and the rebinning formulas described in equation (1) the cone beam projection image $P(\alpha, \beta, q)$ from which the information in this column p was taken in the rebinning and hence which motion state \mathbf{M}' was present can be calculated. With this new more accurate motion state \mathbf{M}' the voxel position \mathbf{x} can be recalculated to $\mathbf{x}'_{\mathbf{M},\theta}$, followed by the calculation of the new position (p', q') in the projection image $P(\theta, p, q)$. This more accurate position (p', q') can then be used to write its values to the voxel \mathbf{x} . The steps 3) to 5) can be repeated to achieve even more accuracy, however experiments have shown that even one iteration is able to yield good results.

For a more efficient implementation, we propose a modification of the algorithm stated above. Instead of using the exact motion state \mathbf{M}' that was calculated for every position p in the projection image $P(\theta, p, q)$, we suggest choosing 3 motion states per projection image, whilst interpolating the motion states in between. Specifically, we chose to use the motion state at position $p = 0$ which corresponds to the initial motion state $\mathbf{M}_{\alpha=\theta}$ as well as one motion state for one column left and one column right of $p = 0$. However, we are not choosing the left and rightmost columns in the projection image $P(\theta, p, q)$, but the columns furthest to the left and right that are actually reached in a projection. These columns need to be calculated individually, as they depend on several parameters. For the three chosen columns the corresponding motion states are calculated using the rebinning equations (1). For a position (p, q) in the projection image $P(\theta, p, q)$ the corresponding motion state is then obtained from the motion states of the two closest chosen columns using linear interpolation of the motion values.

2.2 Motion Compensated Iterative Weighted Filtered Backprojection

The method described above can also be combined with an iterative weighted filtered backprojection algorithm, as the one described by Sunnegårdh et al. [7]. The iterative reconstruction algorithm can be described with the vectors \mathbf{p} and \mathbf{f} denoting the projection data and the resulting image data, and the matrices \mathbf{Q} and \mathbf{P} denoting backprojection and forward projection operators, respectively. In our case, \mathbf{Q} is the weighted filtered backprojection by Stierstorfer et al. [6] and \mathbf{P} is the forward projection by Joseph [8]. The iterative reconstruction process can then be described as

$$\mathbf{f}_{k+1} = \mathbf{f}_k - \alpha \mathbf{Q}(\mathbf{P}\mathbf{f}_k - \mathbf{p}). \quad (2)$$

An initial result $\mathbf{f}_0 = \mathbf{Q}\mathbf{p}$ is calculated by backprojecting the rebinned and filtered projection data. The initial result \mathbf{f}_0 is then forward projected and the difference to the initial projection data \mathbf{p} is calculated. This difference image is backprojected and subtracted from the previous result \mathbf{f}_0 using a factor α . This is repeated multiple times, improving the result with every iteration.

Including the proposed motion compensation method into this iterative reconstruction process consists of two parts. First, the motion compensation is included into the backprojection operator \mathbf{Q} as described above. Second, the motion also needs to be considered in the forward projection operator \mathbf{P} . However, in the forward projection the motion is included again rather than removed, as the forward projected result is compared to the original not motion compensated projection data \mathbf{p} . Including the motion in the forward projection operator \mathbf{P} is performed similar to the motion compensation in the backprojection operator \mathbf{Q} . In the forward projection, we are assuming an implementation that iterates through all positions (p, q) in the projection image $P'(\theta, p, q)$ that should be calculated. For each position (p, q) , the corresponding ray through the voxel volume is calculated and the values along its path are accumulated. This is repeated for all projection images $P'(\theta, p, q)$. Knowing the position (p, q) the corresponding motion state at this position can be interpolated as described above. The motion state can then be included in the calculations of each ray changing its path through the voxel volume accordingly. The value accumulated along the rays path is written to the initial position (p, q) of the projection image $P'(\theta, p, q)$. However, the motion compensation calculations need to be included in every backprojection and every forward projection step of every iteration.

Hence, we propose to use a modification of the iterative reconstruction algorithm used above, as the one described by Sunnegårdh et al. [9], that allows for a more efficient implementation of the proposed motion compensation algorithm. The algorithm stated in equation (2) is adapted by considering the combination of forward projection followed by backprojection to be the sum of a low-pass filter L and a linear operator A that causes artifacts:

$$QP = L + A. \quad (3)$$

Hence the iterative reconstruction equation (2) can be rewritten to

$$f_{k+1} = f_0 - (QP - L) f_k \quad (4)$$

as described in more detail by Sunnegårdh et al. [9]. This modification allows to include the motion compensation algorithm only in the first backprojection step, without the need to repeat the motion compensation calculations in every iteration. Furthermore, there is no need to adapt the forward projection operator to consider motion.

3. RESULTS

To evaluate the proposed motion compensation algorithm the Turbell clock phantom [10] is used, simulating a helical cone beam CT scan, followed by reconstruction with and without motion compensation.

To evaluate the motion compensation the phantom was simulated to be moving along a predefined path. In this case a rigid motion was introduced, moving the entire phantom via translation, rotation, and moving of the rotation center. For the acquisition of each projection image $P(\alpha, \beta, q)$ a motion state M_α of the phantom was defined, consisting of values for translation, rotation, and the position of the rotation center. The values of all motion states M_α used for the simulation are saved for later use in the motion compensated reconstruction as it is assumed throughout this paper that the motion present during acquisition is known.

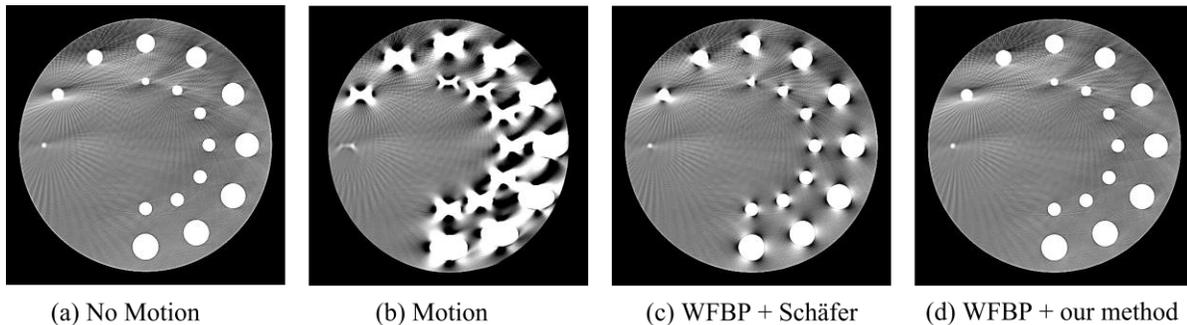


Figure 1. Example reconstructions of the Turbell clock phantom (a) without introducing motion, (b) introducing a translation in x -direction performing a 5mm sinus without motion compensation, (c) introducing the same motion and compensating it with the combination of Schäfer's method and the WFBP algorithm, and (d) introducing the same motion and compensating it with our proposed method. Greyscale window C: 0 HU, W: 100 HU.

Fig. 1 (a) shows the phantom simulated without any motion and reconstructed using the WFBP algorithm [6] as reference. Fig. 1 (b) shows the WFBP reconstruction of the phantom for a simulated scan including motion. The motion included in this example is a translation in x -direction describing a sinus curve with an amplitude of 5mm and one oscillation per scanner rotation. Fig. 1 (c) shows the reconstruction of a scan of the phantom with the same motion, now including the motion compensation method proposed by Schäfer et al. [4] but combined with the WFBP algorithm. Fig. 1 (d) shows the reconstruction result of the scan simulated with the same motion, using our proposed motion compensation method. In this case only one iteration of steps 3) to 5) of our proposed algorithm was performed. It can be seen that for the combination with the WFBP algorithm our proposed method shows significant improvement regarding the motion artifact reduction, when compared to the combination of the WFBP algorithm with Schäfer's method, even if only one iteration of our proposed algorithm is performed.

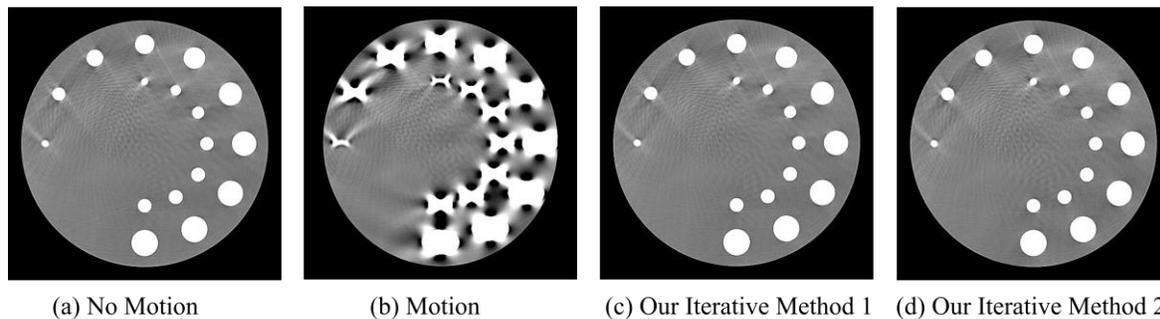


Figure 2. Iterative reconstructions of the Turbell clock phantom (a) without introducing motion, (b) introducing a translation in x-direction performing a 5mm sinus without motion compensation, (c) introducing the same motion and compensating it with the motion being considered in every backprojection and every forward projection, and (d) introducing the same motion and compensating it only in the first backprojection. Greyscale window C: 0 HU, W: 100 HU.

Fig. 2 shows the results for the two motion compensated iterative reconstruction methods. Again Fig. 2 (a) and (b) show the reference images without and with the introduced motion, the same translation introduced in the experiments shown in Fig. 1. Fig. 2 (c) shows the result for the iterative reconstruction where the motion compensation is included in every backprojection and every forward projection step. Fig. 2 (d) shows the iterative reconstruction result, for the version, where the motion compensation is only included in the first backprojection step.

4. CONCLUSION

With their motion compensation algorithm Schäfer et al. [4] proposed a solution for motion artifacts in CT images that yields very good results, if the motion present during acquisition is known. However, they proposed the combination of their motion compensation with the reconstruction algorithm of Feldkamp, Davis, and Kress [5]. Combining their method with reconstruction algorithms containing a rebinning step like the WFBP algorithm [6] does not yield similarly good results as different motion states are mixed.

In this paper we propose an extension of the method by Schäfer et al. that considers the rebinning process in the backprojection step, making a combination with the WFBP reconstruction algorithm possible. Furthermore, we propose the combination of our motion compensation method with iterative weighted filtered backprojection.

Our experiments show promising motion compensation results both with the WFBP algorithm and the iterative reconstruction algorithm that are comparable to the results achieved with the method by Schäfer et al. combined as suggested with the reconstruction algorithm of Feldkamp, Davis, and Kress.

In our experiments we focused on the introduction and compensation of rigid motion, however the proposed algorithm should also work with non-rigid motion, considering the motion states of individual voxels, rather than the entire phantom.

REFERENCES

- [1] J. Hahn, H. Bruder, C. Rohkohl, T. Allmendinger, K. Stierstorfer, T. Flohr, and M. Kachelrieß, "Motion compensation in the region of the coronary arteries based on partial angle reconstructions from short-scan CT data," *Medical physics*, 44(11), 2017.
- [2] R. Bhagalia, J. D. Pack, J. V. Miller, and M. Iatrou, "Nonrigid registration-based coronary artery motion correction for cardiac computed tomography," *Medical physics*, 39(7Part1), pp. 4245-4254, 2012.

- [3] H. Bruder, C. Rohkohl, K. Stierstorfer, and T. Flohr, "Compensation of skull motion and breathing motion in CT using data-based and image-based metrics, respectively," *Proc. SPIE 9783, Medical Imaging 2016: Physics of Medical Imaging*, 97831E, 22 March 2016.
- [4] D. Schafer, J. Borgert, V. Rasche, and M. Grass, "Motion-compensated and gated cone beam filtered back-projection for 3-D rotational X-ray angiography," *IEEE Transactions on Medical Imaging*, vol. 25, no. 7, pp. 898-906, July 2006.
- [5] L. A. Feldkamp, L. C. Davis, and J. W. Kress, "Practical cone-beam algorithm," *J. Opt. Soc. Am. A* 1, pp. 612-619, 1984.
- [6] K. Stierstorfer, A. Rauscher, J. Boese, H. Bruder, S. Schaller, and T. Flohr, "Weighted FBP—a simple approximate 3D FBP algorithm for multislice spiral CT with good dose usage for arbitrary pitch," *Physics in Medicine & Biology*, 49(11), p. 2209, 2004.
- [7] J. Sunnegardh, and P.E. Danielsson, "Regularized iterative weighted filtered backprojection for helical cone-beam CT," *Medical physics*, 35(9), pp. 4173-4185, 2008.
- [8] P.M. Joseph, "An improved algorithm for reprojecting rays through pixel images," *IEEE transactions on medical imaging*, 1(3), pp. 192-196, 1982.
- [9] J. Sunnegardh and K. Stierstorfer, "A New Method for Windmill Artifact Reduction," *The 12th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, pp. 420-423, 2013.
- [10] H. Turbell, "Cone-beam reconstruction using filtered backprojection," *Diss. Linköping University Electronic Press*, 2001.