Projection Selection Algorithms for Discrete Tomography

László Varga*, Péter Balázs**, and Antal Nagy

Department of Image Processing and Computer Graphics
University of Szeged
Árpád tér 2, H-6720 Szeged, Hungary
{vargalg,pbalazs,nagya}@inf.u-szeged.hu

Abstract. In this paper we study how the choice of projection angles affect the quality of the discrete tomographic reconstruction of an object. We supply four different strategies for selecting projection angle sets and compare them by conducting experiments on a set of software phantoms. We also discuss some consequences of our observations. Furthermore, we introduce a possible application of the proposed angle selection algorithms as well.

Keywords: discrete tomography, reconstruction, adaptive projection acquisition, non-destructive testing.

1 Introduction

The main goal of tomographic reconstruction is to determine the inner structure of objects from their projections taken along a set of directions. This is usually done by exposing the objects to some electromagnetic or particle radiation on one side, measuring the amount of transmitted radiation on the other side, and reconstructing the inner densities of the objects by a suitable algorithm.

There are several appropriate algorithms for tomographic reconstruction capable of giving accurate results when sufficiently many projections are available. However the cost of acquiring projections can be extremely high, and the radiation can damage the objects of study, too. Due to these problems there is always a need to reduce the number of projections required for the reconstruction.

For this purpose, one can develop more accurate new algorithms by using some a priori information of the objects of interest, e.g., by assuming that the objects can only consist of a few different materials with known attenuation coefficients, and/or their shape fulfill some special property [3,4]. Another approach for reducing the number of required projections is to try to take the projections with

* Corresponding author.

** This research was partially supported by the TÁMOP-4.2.2/08/1/2008-0008 and TÁMOP-4.2.1/B-09/1/KONV-2010-0005 programs of the Hungarian National Development Agency and by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences.

J. Blanc-Talon et al. (Eds.): ACIVS 2010, Part I, LNCS 6474, pp. 390–401, 2010.

the highest information content. Thus, a smaller set of projections can hold just enough information for a proper reconstruction. In [8] the authors showed that in the case of continuous reconstruction the number of required projections can be significantly reduced by choosing the right angles with some heuristic algorithms. It was also shown in [7,10] that the choice of the proper projection angles can particularly be crucial in the case of discrete tomography, when usually only a handful of projections are available for the reconstruction.

This paper focuses on examining the differences between the discrete tomographic reconstructions of the same object performed by the same algorithm but from different projection sets. We introduce several algorithms for finding the appropriate projection angles of given objects, and compare these approaches through experiments conducted on a set of software phantoms. We discuss a possible practical application of the proposed algorithms as well.

The structure of the paper is the following. In Section 2 we explain the central problem of discrete tomography and describe an algorithm for solving it. In Section 3 we introduce several methods to select proper angles for the reconstruction. In Section 4 we give the details of the frameset applied to compare our angle selection algorithms. In Section 5 we summarize the experimental results. In Section 6 we discuss a possible application of the proposed angle selection algorithms. Finally, in Section 7 we summarize our observations.

2 Discrete Tomographic Reconstruction

In a formulation of two dimensional transmission tomography the goal is to reconstruct an unknown $f(u, v)$ function from a set of its projections given by

$$[Rf](\alpha, t) = \int_{-\infty}^{\infty} f(t\cos(\alpha) - q\sin(\alpha), t\sin(\alpha) + q\cos(\alpha))\ dq \tag{1}$$

line integrals with different α angles and t distances from the origin. Although there is a thorough mathematical theory and an exact formula for continuous reconstruction when all the possible projections are available, in a practical application we can only handle a finite number of projections and the function itself must be also discretised. In the sequel we will assume that the function $f : \mathbb{R}^2 \to \{0, 1\}$ to be reconstructed is binary and takes a constant value on each unit square determined by the 2-dimensional integer lattice, that is

$$f(u + a, v + b) = f(u + c, v + d),\ u, v \in \mathbb{Z},\ a, b, c, d \in [0, 1). \tag{2}$$

We will further assume that the f function has a bounded support. With these restrictions the final task can be regarded as the reconstruction of a two dimensional binary image (where 0 stands for the background and 1 for the foreground) and the reconstruction problem can be represented by a system of equations

$$\mathbf{Ax} = \mathbf{b}, \quad \mathbf{A} = (a_{i,j})_{n\times m} \in \mathbb{R}^{n\times m},\ \mathbf{x} \in \{0, 1\}^n,\ \mathbf{b} \in \mathbb{R}^m\ . \tag{3}$$

Here, $\mathbf{x}$ is the vector of all n pixels of the unknown image to be reconstructed, $\mathbf{b}$ is the vector of the total of m measured projection values and $\mathbf{A}$ describes the

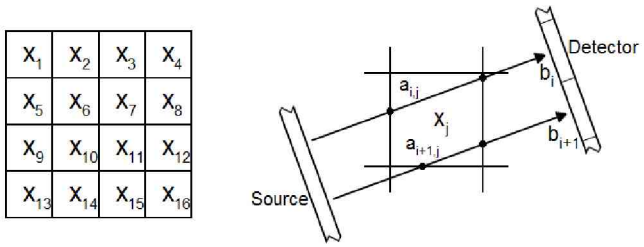

Fig. 1. Representation of the ordering of the pixels and the parallel beam geometry used

connection between the pixels and the projection beams by all $a_{i,j}$ giving the length of the line segment of the i-th ray through the j-th pixel. Figure 1 shows an example for the projection representation we used. Now, the task is to solve (3) which is usually performed by the different versions of the so-called algebraic reconstruction technique (see [1,3,5]).

In our case we used a slightly different approach that reformulates the reconstruction as an energy minimization problem. The algorithm given in [9] applies D.C. programming (a technique for minimizing the difference of convex functions) for minimizing the energy function given as

$$J_\mu(\mathbf{x}) := \|\mathbf{Ax}-\mathbf{b}\|_2^2 + \frac{\gamma}{2}\sum_{j=1}^{n}\sum_{l\in N_4(j)}(\mathbf{x}_j-\mathbf{x}_l)^2 - \mu\frac{1}{2}\langle \mathbf{x}, \mathbf{x}-\mathbf{e}\rangle\,, \quad \mathbf{x}\in[0,1]^n\,, \quad (4)$$

where γ is a given constant controlling the weight of the smoothness term on the image, $N_4(j)$ is the set of pixels 4-connected to the j-th pixel, and $\mathbf{e}$ denotes the vector with all n^2 coordinates equal to 1. Minimizing the energy function (4) with the proper parameters and projections, results in an $\mathbf{x}$ vector which contains the pixel values of the reconstruction. The main advantage of this approach is that it can handle the possible inconsistency of the equation system (3) and it incorporates prior information into the model, too.

The basic operation of this algorithm starts out by looking for a continuous result by minimizing the $J_\mu(\mathbf{x})$ function omitting the last term (by setting $\mu = 0$). In the proceeding we iteratively increase μ with a $\mu_\Delta > 0$ value by which – as it can easily be proved – the algorithm will converge to a binary result. The exact description of the algorithm and its mathematical background can be found in [9]. In the sequel we will refer to this algorithm as "DC".

3 Angle Selection Algorithms

It has already been shown in [7,10] that the choice of projection angles can have a significant influence on the result of the reconstruction of certain objects. However, the previous studies dealt only with the case when the projections were taken equiangularly. Here, we extend those results to the case when the projections can be taken from arbitrary directions around the objects.

While conducting experiments with equiangular projection sets – as it was done in [10] – is relatively easy, we found that the problem becomes much more complex when we omit the assumption of equiangularity. Even if we restrict ourselves to angles of integer degrees between 0° and 179° the solution space can be too large to perform an exhaustive search in it. Therefore, we will use heuristic algorithms for finding not necessarily the best, but sufficiently good angle sets to the reconstruction.

In the following, $S(\alpha, p)$ will denote a set of angles defined as

$$S(\alpha, p) = \{\alpha + i\frac{180^\circ}{p} \mid \ i = 0, \ldots, p-1\} \ , \tag{5}$$

for a p number of projections and a starting angle α. On the other hand, L will stand for an ordered list of angles $L = \langle \alpha_1, \alpha_2, \ldots, \alpha_p \rangle$ with arbitrary α_i-s and arbitrary p. We will use the notations $\mathbf{x}_{S(\alpha,p)}$ and $\mathbf{x}_L$ for the reconstructions from the projections taken with $S(\alpha, p)$ and L angle sets, respectively, produced by the DC reconstruction algorithm. Finally, we will denote by $\mathbf{x}^*$ the vector of pixel values of the expected reconstruction (the image of the original object) and use $RME(\mathbf{x}^*, \mathbf{y})$ for measuring the relative mean error of a given $\mathbf{y}$ vector of reconstructed pixels, calculated as

$$RME(\mathbf{x}^*, \mathbf{y}) = \frac{\sum_{i=1}^{n} |x_i^* - y_i|}{\sum_{i=1}^{n} x_i^*}. \tag{6}$$

Informally, the RME value is the number of mistaken pixels in the reconstruction normalized with the number of object pixels of the original image.

We consider only integer angles between 0° and 179°. In our notation 0° stands for the projection with vertical beams aimed from the bottom to the top of the image and increasing degrees of angles means a counterclockwise rotation.

We now present four different angle selection methods, two simple equiangular ones for a basic reference, and two more complex ones allowed to result in arbitrary angles from the defined possibilities.

Naive angle selection

The method we call **Naive** is a simple, picture independent angle set selection technique. For every image and a given projection number p we simply choose the appropriate equiangular projection set with 0° starting angle defined by $S(0^\circ, p)$. Note that most applications use a similar approach for choosing angles.

Equiangular search

We can improve the results of the naive angle selection by trying to make a reconstruction of the phantoms from the $S(\alpha, p)$ projection sets with all integer α starting angles varying form 0° to $\lfloor \frac{180}{p} \rfloor^\circ$ and selecting the projection set producing the smallest $RME(\mathbf{x}^*, \mathbf{x}_{S(\alpha,p)})$ value. This is exactly the same algorithm that was used in [10] for similar experiments. In the sequel we will simply call this algorithm **EquiAng**.

Greedy angle testing

One of our non-equiangular approaches for optimal angle selection is a greedy algorithm. The basic idea of this method is to start out with a projection angle set containing one single predetermined angle, and then to add new angles iteratively to the current set, based on a local decision. In every iterations we add the angle to the current set which causes the greatest improvement in the resulting reconstruction. The best angle is chosen by trying to add each unused projection angle to the original set, making a reconstruction with the new set, and keeping the angles belonging to the reconstruction with the smallest RME value. At the end of this algorithm the result is an ordered list of angles with decreasing significance.

Greedy: Greedy angle selection algorithm.
Input: $\mathbf{x}^*$ vector of image pixel values, $k \geq 2$ maximal number of angles, and α_1 predetermined angle.
Output: $L = \langle \alpha_1, \alpha_2, \ldots, \alpha_l \rangle$ angle list so that $l \leq k$.

Step 1. Set $L_1 = \langle \alpha_1 \rangle$, $i = 1$;
Step 2. Let $i \leftarrow i + 1$;
Step 3. Let $0^\circ \leq \alpha^* \leq 179^\circ$ be an integer angle for which $RME(\mathbf{x}^*, \mathbf{x}_{\langle L_{i-1}, \alpha^* \rangle})$ is minimal;
Step 4. Let the next list of angles be $L_i = \langle L_{i-1}, \alpha^* \rangle$;
Step 5. If $i = k$ or $RME(\mathbf{x}^*, \mathbf{x}_{L_i}) = 0$ return with L_i otherwise go to **Step 2**

In addition to the image data and the maximal number of projection angles this method also requires an α_1 starting angle since our implementation of the DC reconstruction algorithm cannot produce a reasonable result with just one projection. In our experiments for each image this angle was given as an integer angle with minimal $\|\mathbf{x}^* - \mathbf{y}_\alpha\|_2^2$ value, where $\mathbf{y}_\alpha$ stands for the reconstruction of the image produced by the SIRT algorithm (described in [5]) from only one projection with α angle.

Altering angles with simulated annealing

This non-equiangular algorithm is based on the fact that the optimal angle searching problem can itself be represented as an energy minimization task. Having this in mind, the optimal L^* list of angles must satisfy

$$RME(\mathbf{x}^*, \mathbf{x}_{L^*}) = \min_L \; RME(\mathbf{x}^*, \mathbf{x}_L). \tag{7}$$

In this way we can reformulate the optimal angle searching problem as finding the minimal value of $RME(\mathbf{x}^*, \mathbf{x}_L)$ with varying L. Because of the complexity of the new energy function – as it contains the result of a reconstruction – we have chosen to do the minimization with simulated annealing [6].

AltAng: Angle selection with simulated annealing.
Input: $\mathbf{x}^*$ vector of image pixel values, $L_0 = \langle \alpha_1, \alpha_2, \ldots, \alpha_p \rangle$ starting angle list, T_0 starting temperature, $0 < h < 1$ cooling rate, N angle step range.
Output: $L = \langle \alpha'_1, \alpha'_2, \ldots, \alpha'_p \rangle$.

Step 1 Choose a random $i \in \{1, \ldots, p\}$ integer from a uniform distribution;
Step 2 Choose a random $\alpha'_i \in \{\alpha_i - N, \ldots, \alpha_i + N\} \setminus \{\alpha_i\}$ integer angle from a uniform distribution, and replace the i-th angle so that the new angle list is $L' = \langle \alpha_1, \ldots, \alpha_{i-1}, \alpha'_i, \alpha_{i+1}, \ldots, \alpha_p \rangle$;
Step 3 If $RME(\mathbf{x}^*, \mathbf{x}_{L'}) < RME(\mathbf{x}^*, \mathbf{x}_L)$ then $L \leftarrow L'$ and proceed with **Step 5**;
Step 4 Let $L \leftarrow L'$ with $e^{-\frac{RME(\mathbf{x}^*, \mathbf{x}_{L'}) - RME(\mathbf{x}^*, \mathbf{x}_L)}{T}}$ probability;
Step 5 If a stopping criteria is met then return with the current L angle set, otherwise let $T \leftarrow hT$ and start over from **Step 1**;

In a more informal description, this algorithm starts with a basic fixed-size angle list and in each iteration it tries to alter one of the angles changing it inside a predefined neighborhood. If the random change results in a better angle set then we accept the new angle. If the new angle is worse than the previous one we can accept it with a probability driven by the T temperature and the difference between the RME values of the two reconstructions with the different angle sets. We decrease the temperature in each iteration and – by this – the probability of accepting worse angles is also decreased.

4 Test Frameset and Phantom Images

To compare the performance of the given algorithms we performed experimental tests on a set of software phantoms. In this section we will give a description of the test frame set and provide more details on the algorithms given in Section 3.

We conducted the experiments with five software phantoms of different types, all of them were scaled to the same size of 256 by 256 pixels. These phantoms can be seen in Figure 2.

As described before we used parallel beam geometry with each measured value representing the projection of the image along one single line. In every projection we set the distance of the beams – or detector elements – to 1 pixel and used as many as needed to cover the whole image. The rotation center of the angles – or the origin of the coordinate system belonging to Equation (1) – was placed exactly into the center of the image. The distances of the projection lines from the origin were set $i + 1/2$ pixels with integer i values.

The reconstructions were performed by the DC algorithm implemented in C++ with GPU acceleration using the CUDA sdk (see [13]). In each reconstruction we set the parameters of the algorithm as described in [12] except the setting of Δ_μ directly to 0.1 instead of calculating it dynamically for every reconstruction. In this way we could perform one reconstruction in 10-30 seconds,

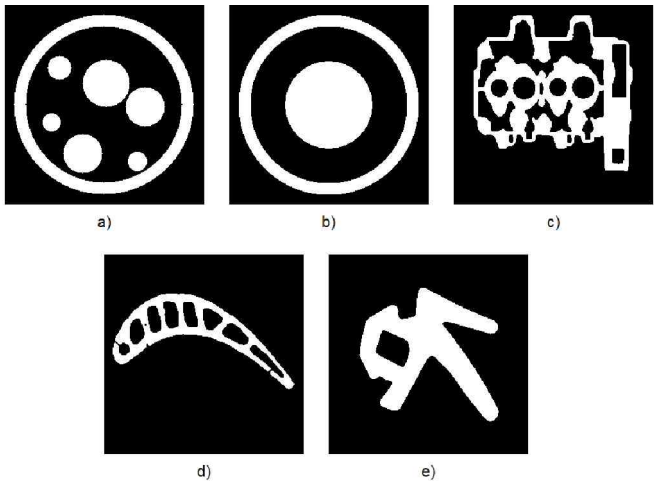

Fig. 2. Phantom images used in the experiments

depending on the complexity of the image to be reconstructed and the number of projections used.

For each phantom we used the angle selection algorithms for finding angle sets with 2 to 11 projections. In the case of the **Naive** and **EquiAng** algorithms this meant ten processes for each phantom with different p projection numbers. In the case of the **AltAng** algorithm we took the best angle set from five different runs for every phantom and number of projections. With the **Greedy** algorithm we only had to perform one process with a $p = 11$ value for each phantom. This made a total of 35320 reconstructions performed in about 200 hours on an Intel Core 2 CPU and an NVIDIA Geforce 8800 GT GPU.

The parameters of of the **AltAng** algorithm were set empirically. We set the values $T_0 = 0.02$ and $h = 0.95$. The neighborhood of the projection angles was determined depending on the projection number p as $N = (\lfloor 180/p \rfloor - 5)^\circ$, that is, we decreases the neighborhood of the angles when there were more projections, in order to keep the probability of changing the order of the angles low. It is important to note that the search space of possible solutions is not reduced this way. For each image and p number of projections the starting angle set of this algorithm was the output of the **EquiAng** algorithm, i.e., the $S_{\alpha,p}$ angle set that produced the best reconstruction. Each time the stopping criteria for the process was reaching 200 iterations, or a perfect reconstruction. When the equiangular search already resulted in a projection set giving a perfect reconstruction we did not start the random search at all.

5 Numerical Results and Their Interpretation

After performing the tests we compared the angle sets produced by the four angle selection algorithms. Table 1 contains a summary of the results by giving

the RME values of the reconstructions with the angle sets of the different algorithms. In addition, Figure 3 presents the same result graphically for the phantom of Figure 2c.

Our first consideration was to compare the results of the different angle selection algorithms on the same phantoms with the same numbers of projections. As it was already shown in [7,10] the **EquiAng** algorithm usually gives better – but never worse – results than the **Naive** method. This is not so surprising since the equiangular search includes the angle set of the **Naive** algorithm.

We can also make valuable observations on the results of the **Greedy** algorithm. For certain images the angles given by this method can be significantly better than the ones given by the **EquiAng** or the **Naive** approaches (see the results for Figures 2d-e). However, in the case of the rotation invariant phantoms the greedy search can result in a local minima of RME. From the entries of

Table 1. Numerical results giving the RME values of the reconstructions produced by the DC algorithm with the angle sets proposed by different angle selection strategies (columns indicate the numbers of projections, and rows give the applied angle selection algorithm). Tests belonging to blank cells were not performed for reasons given in Sections 3 and 4.

Proj.Num.	2	3	4	5	6	7	8	9	10	11
Figure 2a										
Naive	1.1218	0.7943	0.5549	0.2904	0.0087	0	0	0	0	0
EquiAng	0.9339	0.6926	0.4687	0.1687	0.0002	0	0	0	0	0
Greedy	0.8568	0.5888	0.4664	0.2650	0.0407	0				
AltAng	0.0842	0.5749	0.3856	0.1082	0					
Figure 2b										
Naive	0.6586	0.4162	0	0	0	0	0	0	0	0
EquiAng	0.6070	0.4119	0	0	0	0	0	0	0	0
Greedy	0.6357	0.4062	0.2665	0.0707	0					
AltAng	0.6051	0.2687								
Figure 2c										
Naive	0.7951	0.7245	0.5392	0.5546	0.3465	0.4596	0.2725	0.1650	0.0643	0.0926
EquiAng	0.7951	0.7245	0.5146	0.4438	0.3362	0.3517	0.2725	0.1643	0.0643	0.0409
Greedy	0.7435	0.5912	0.4513	0.3736	0.2929	0.2090	0.1544	0.1057	0.0318	0
AltAng	0.7380	0.5709	0.4432	0.3450	0.2785	0.1841	0.1009	0.0444	0.0071	0
Figure 2d										
Naive	1.0348	0.7544	0.6239	0.3900	0.1615	0.0019	0	0	0	0
EquiAng	1.0348	0.7153	0.5308	0.2709	0.0849	0	0	0	0	0
Greedy	1.0751	0.7440	0.3672	0.1239	0.0508	0.0005	0			
AltAng	0.8447	0.5974	0.4091	0.1128	0.0044					
Figure 2e										
Naive	0.7742	0.5836	0.4255	0.1512	0	0	0	0	0	0
EquiAng	0.5475	0.1691	0.0500	0.0001	0	0	0	0	0	0
Greedy	0.5756	0.1530	0.0117	0						
AltAng	0.5268	0.1617	0.0029	0						

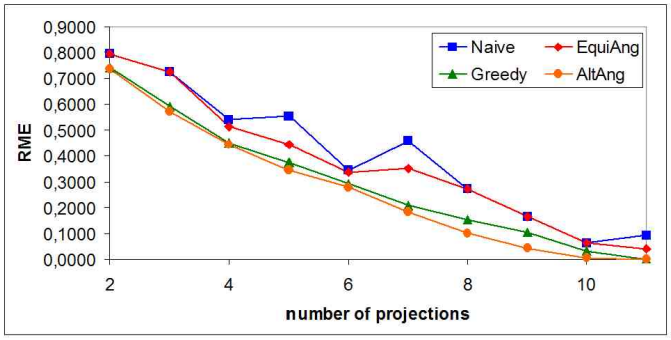

Fig. 3. *RME* values of the reconstructions of the phantom in Figure 2c with the angle sets provided by the four different angle selection algorithms

Table 1 we can also deduce that for the two phantoms with the rings (the ones in Figures 2a and 2b) the greedy search cannot yield a proper reconstruction from even the same number of projections sufficient for the **Naive** approach.

The final algorithm to compare is the simulated annealing based one. As we can deduce from Table 1, this approach gave in most of the cases better results than the others. Compared to the **Naive** or the **EquiAng** algorithms this behavior of algorithm **AltAng** is quite obvious since the base of the search in **AltAng** is given by those methods and the relatively small starting temperature does not allow acceptance of angle sets much worse than the previous ones. Even comparing the results to the ones of the **Greedy** algorithm, we can say that **AltAng** seems to give the best angle sets. The explanation of this is that this approach does not suffer from the defects of the other algorithms, i.e., it does not restrict the search to equiangular projections sets, and does not make unchangeable local decisions in the computation.

Despite the good results of the simulated annealing based strategy we must highlight that it has a serious drawback in making random searches. This process is greatly driven by fortune, and we cannot hope to get always such good results. This is why we decided here to take the best results from five different runs for each phantom and projection number. Naturally running the algorithm more times or allowing more iterations with a higher starting temperature could produce more stable results, probably closer to the optimal projection angle set.

As a summation of the comparison we can say that the result of a binary reconstruction can significantly depend on the angles chosen for creating projections of the objects of study. That is, we can get entirely different reconstruction results even if we use equiangular angle sets for the projections. If we allow the projections to be taken in arbitrary directions the difference can be even more significant. In a practical application this means that we can get acceptable results from fewer projections, and thus reduce the required number of projections – and with it the cost and the amount of radiation – by selecting the right angles. As a demonstration of our results Figure 4 shows an example with the

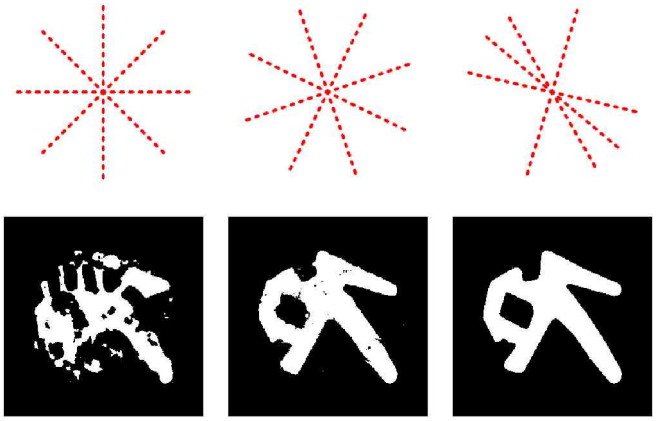

Fig. 4. Reconstructions of the phantom in Figure 2e with $S(0^\circ, 4)$, $S(19^\circ, 4)$ and $L = \langle 30^\circ, 50^\circ, 77^\circ, 163^\circ \rangle$ projection sets respectively. (Red dashed lines indicate the directions of the projections, images below are the corresponding reconstructions.)

reconstruction of the phantom of Figure 2e from three different projection sets with the same number of projections.

We can also come to further conclusions based on the data in Table 1. First, the result of a reconstruction can significantly be improved by choosing the angles in arbitrary directions. Another important consequence is that the accuracy of the discrete tomographic reconstruction algorithms can only be compared for fixed projection sets, and evaluations of reconstruction algorithms in the literature should also contain a detailed explanation of the applied projection geometry.

6 An Application of the Angle Selection Algorithms

Based on our results we can easily think of possible practical applications of the angle selection algorithms introduced in Section 3 as well. One such application has already been given in [10], by proposing to use a blueprint for determining the right angles to take projections from, in nondestructive testing (see [2]).

The paper [10] discussed only equiangular projections. Based on the results presented here further improvement can be achieved by allowing arbitrary angles to be used. Of course, we usually cannot put the object of study with the same exact direction into the scanner and acquire the desired projections. Thus the angles can slightly differ from the perfect ones. It can be useful to take a look on what happens if we alter the angles found by our algorithms. We conducted further experiments by rotating the best angle sets we found by angles $-90^\circ, -89^\circ, \ldots, 89^\circ$. Some examples for the results can be seen graphically in Figure 5 for Figure 2d (left) and Figure 2e (right).

From Figure 5 we can deduce that – except a small noise – the curves are relatively smooth, but they increase with a high gradient when getting farther

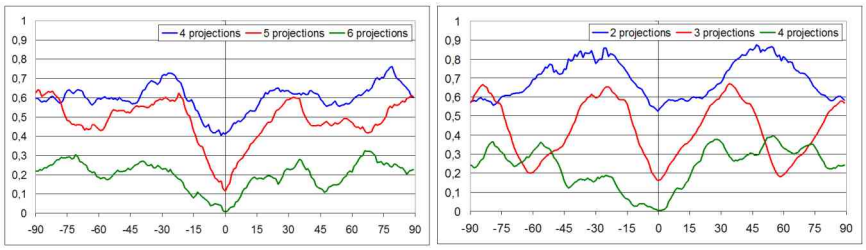

Fig. 5. The graph of RME values after the rotation of the best angle set around the phantoms in Figure 2d (left) and Figure 2e (right) depending on the angle of rotation

from the originally determined angles. This fact suggest that, in a real world application with non-equiangular projections, the objects of study must be placed into the scanner with high precision in order to get a good result without making additional projections. The explanation of this strictness in the angles can be that we developed our algorithms to try to find the angles giving the best reconstruction and altering those angles all at once can easily degrade the quality of the reconstructed image. Nevertheless, we may have a small freedom without getting unacceptable results.

Our observations show that the benefit of finding the right angles is the bigger if the object to be reconstructed is topologically simple. Our results show, that more complex objects — those require several projections for an acceptable reconstruction — are less dependent on the choice of projection angles. With an increasing number of projections, the differences between these projections get negligible and we can not get an improvement worth noting only by getting the better angles. Therefore we would advise applying the angle selection algorithm described in Section 3 for practical applications, in the case of simple objects.

7 Conclusion

In this paper we examined how the choice of the angles of projections can affect the result of binary tomographic reconstruction algorithms. We presented four different angle selection strategies and compared them through tests performed on a set of software phantoms. We found that the selection of angles can significantly affect the result of the DC tomographic reconstruction algorithm. We also showed that such results can lead to further conclusions and we also proposed a possible application for the presented algorithms, too.

In the future we plan to extend our studies to other reconstruction algorithms, and also the case when the measured data is affected by noise, which is the common situation in real applications.[1] In addition providing a more theoretical explanation of the problem would be a major breakthrough as well.

[1] By the time of reviewing the current paper we have already made efforts in this direction and submitted [11].

Acknowledgments

The authors would like to thank Joost Batenburg for providing test images (Figures 2c and 2d) for the studies. Phantom of Figure 2e was taken from the image database of the IAPR Technical Committee on Discrete Geometry (TC18).

References

1. Batenburg, K.J., Sijbers, J.: DART: a fast heuristic algebraic reconstruction algorithm for discrete tomography. In: IEEE Conference on Image Processing IV, pp. 133–136 (2007)
2. Baumann, J., Kiss, Z., Krimmel, S., Kuba, A., Nagy, A., Rodek, L., Schillinger, B., Stephan, J.: Discrete tomography methods for nondestructive testing. In: [4], ch. 14, pp. 303–331 (2007)
3. Herman, G.T., Kuba, A. (eds.): Discrete Tomography: Foundations, Algorithms and Applications. Birkhäuser, Basel (1999)
4. Herman, G.T., Kuba, A. (eds.): Advances in Discrete Tomography and Its Applications. Birkhäuser, Basel (2007)
5. Kak, A.C., Slaney, M.: Principles of Computerized Tomographic Imaging. IEEE Press, New York (1999)
6. Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., Teller, E.: Equation of state calculation by fast computing machines. J. Chem. Phys. 21, 1087–1092 (1953)
7. Nagy, A., Kuba, A.: Reconstruction of binary matrices from fan-beam projections. Acta Cybernetica 17(2), 359–385 (2005)
8. Placidi, G., Alecci, M., Sotgiu, A.: Theory of adaptive acquisition method for image reconstruction from projections and application to EPR imaging. Journal of Magnetic Resonance, Series B, 50–57 (1995)
9. Schüle, T., Schnörr, C., Weber, S., Hornegger, J.: Discrete tomography by convex-concave regularization and D.C. programming. Discrete Applied Mathematics 151, 229–243 (2005)
10. Varga, L., Balázs, P., Nagy, A.: Direction-dependency of a binary tomographic reconstruction algorithm. In: Barneva, R.P., Brimkov, V.E., Hauptman, H.A., Natal Jorge, R.M., Tavares, J.M.R.S. (eds.) Computational Modeling of Objects Represented in Images. LNCS, vol. 6026, pp. 242–253. Springer, Heidelberg (2010)
11. Varga, L., Balázs, P., Nagy, A.: Direction-dependency of binary tomographic reconstruction algorithms. Submitted to Graphical Models (CompIMAGE 2010 special issue)
12. Weber, S., Nagy, A., Schüle, T., Schnörr, C., Kuba, A.: A benchmark evaluation of large-scale optimization approaches to binary tomography. In: Kuba, A., Nyúl, L.G., Palágyi, K. (eds.) DGCI 2006. LNCS, vol. 4245, pp. 146–156. Springer, Heidelberg (2006)
13. NVIDIA CUDA technology, web page (August 24, 2010), `http://www.nvidia.co.uk/object/cuda_home_new_uk.html`