

Simulating Annealing Approach for Discrete Tomography from Orthogonal Projections Using Boundary Points

Narender Kumar^{a*}, Roopa Kumari^b

^aDepartment of Computer Science and Engineering , HNB, Garhwal University Srinagar Garhwal, Uttarakhand, India

^{ab}Department of Information Technology , HNB, Garhwal University Srinagar Garhwal, Uttarakhand, India

Abstract: In this paper simulating annealing technique is used to minimize the number of solutions from orthogonal projections. Convexity is prior information about the object geometry in the discrete tomography. This information may be useful for reconstruction of binary matrix or binary image from their projections. Boundary point switching is used to find approximate solution. This technique gives better result as compare to general switching.

Key words: Discrete Tomography, Simulating Annealing, Reconstruction, Constraints, Convexity, Boundary point, Switching etc.

I. INTRODUCTION

Image reconstruction from projection or computerized tomography is the technique to find out the density distribution within a physical object from a multiple projections. In computerized tomography we attempt to reconstruct a density function $f(x)$ in R^2 or R^3 from knowledge of its line integral or weighted line sum [1]. This line integral or weighted line sum is the projection of $f(x)$ along line L . The object from the mathematical point of view, corresponds to a density function for which integral or summation in the form of projection data is known. So we can categorize tomography into continuous tomography and discrete tomography. In case of continuous tomography we

consider that both the domain and the range of function are continuous. But in discrete tomography the domain of the function could be either continuous or discrete and the range of the function is finite set of real number. Discrete tomography theory is described by Kuba and Herman [13].

Discrete tomography is used when only few projections for reconstruction are available. But since projection data may be less than the number of unknown variable the problem becomes ill posed. According to Hadmard [2] mathematically problem are termed well posed if they fulfill the following criteria (i) A solution exists (ii) the solution is unique (iii) the solution depend continuously on the data continuously

On the opposite problems that do not meet these criteria are called ill posed. Image reconstruction from few projections is also ill posed problem because it generates a large number of solutions. To minimize number of solution we require some a-priori information about the object geometry. This a-priori information about the object is also called additional constraints on the space of solution. Examples of these are connectivity, convexity [5]-[9] and periodicity [3][14]

Ryser [10] and Gale[11] independently derived necessary and sufficient conditions for the existence of binary matrices from horizontal and vertical projections. Ryser also provided a polynomial time algorithm for finding such a binary matrices.

In this paper the convexity constraint is used to minimize the solution space. The class of *hv* convex binary images has been considered, as the solution space for a given set of projections, the projection set with two orthogonal directions namely, horizontal(1, 0) and vertical direction (0,1) only has been

taken. Thus we have the projection set $\mathcal{P}(R,C)$, $R = \{r_1, r_2, \dots, r_m\}$, and $C = \{c_1, c_2, \dots, c_n\}$. The *hv* convexity with projections from two directions does not provide unique solution [14].

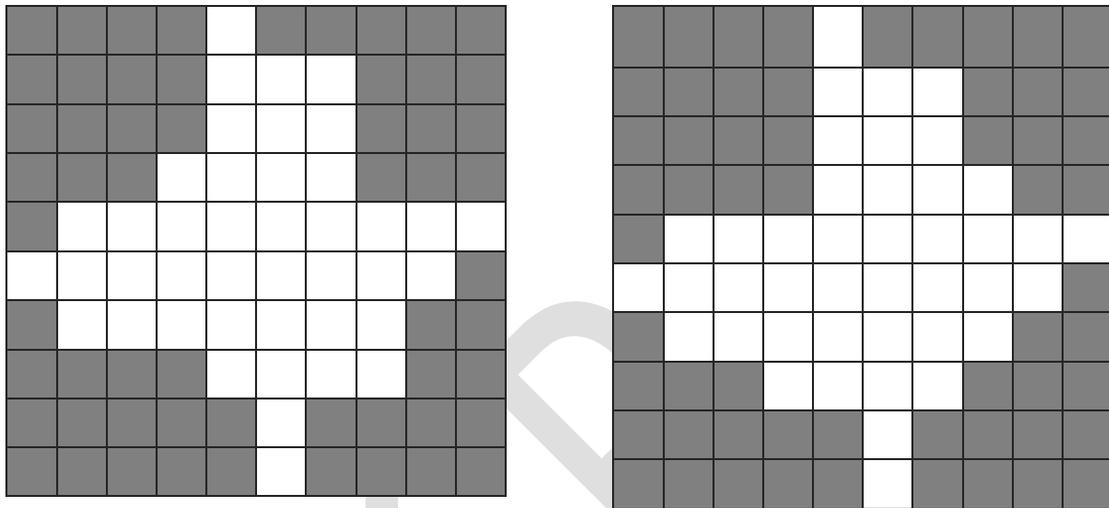


Figure 1: Two different *hv*-convex images with same projection set

In practical applications the class of h-v convex matrices with given projections in two directions $v_1 = (1, 0)$ and $v_2 = (0, 1)$ is very large, making it near impossible to find the exact solution [15]. Moreover many times the exact solution is not required (for example, it may add instrumentation error in projections to the solution), thus optimal solution is the sought after solution in such problems. Since the class is large and the optimization criteria is depending on only projection data and a-priori information usual optimization methods may not work well, hence stochastic optimization technique (minimization algorithm) known as simulated annealing derived from thermodynamics [16] is used here.

Simulated annealing is a global optimization algorithm in the class of stochastic optimization algorithms and Meta Heuristic algorithms. Simulated annealing is an adaption of Metropolis

Hastings, Monte-Carlo algorithm [17] and is used in function optimization. This approach gives a basis for large variety of extensions and specification of general method.

Simulated annealing is inspired by the process of annealing in metallurgy. In this natural process a material is heated and slowly cooled in controlled conditions, so that the size of crystals in material be increased and their defects be reduced, thus improving the strength and durability of material. In this process heat increases the energy of atoms and allows them to move freely from local minimum energy, controlled and slow cooling schedule gives a low energy configuration (lower than initial / previous one) to be found and exploited to make larger crystals with reduced defects.

Since in the materials all atoms are similar, and their behavior is random, thus they form a large search space for finding minimum, the simulated annealing approach gives a generic

stochastic method to find a good approximate global optimum of a function in large search space. This method is more applicable when search space is discrete [16][18].

The analogy of this physical process of annealing in discrete tomography reconstruction problem is as follows:

The cells $(i, j) \in X$ are the state variables (atoms) in the system, which is to be reached at thermodynamic equilibrium at a given temperature, this is represented as each image has a given projection set and/or some a-prior information is also to be satisfied. By slow cooling, the system can be frozen at minimal energy, which is minimization of some cost (fitness) function. In other words simulated annealing based approach can be defined as:

For a given projection set \mathcal{P} and constraints, define the class \mathcal{F} of possible solutions, this class is the search space. Each element F of search space \mathcal{F} is a binary image containing (mn) values (pixels / cells f_{ij}) which are atoms, the heating process says that the values of f_{ij} can be interchanged such that $F' \in \mathcal{F}$ (movement of atoms), cooling will say to find such F' which lowers the energy (cost/ fitness function)

by freezing we understand to reach an optimal or near optimal solution.

In this simulated annealing process the movements of atoms is taken as switching operation, the switching operation as free movement has been considered in [19][20]. In present work the switching is carried on only at boundary of image, since the number of pixels (cells) on boundary is less than the number of pixels (cells) in entire image, this will speed up the complete process..

II. PRELIMINARY

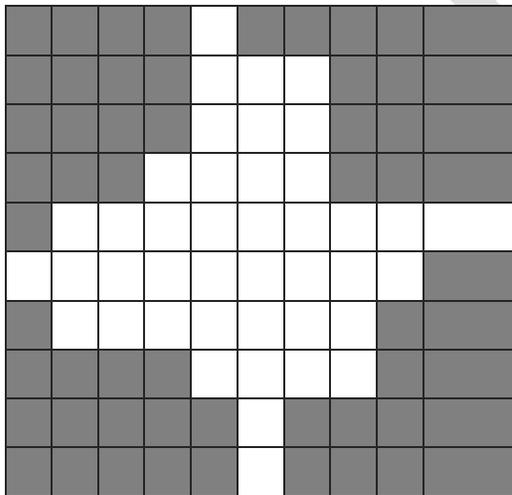
In this section some definition are given which explain the movement of atoms in simulated annealing process.

2.1 Adjacent cells and adjacent points

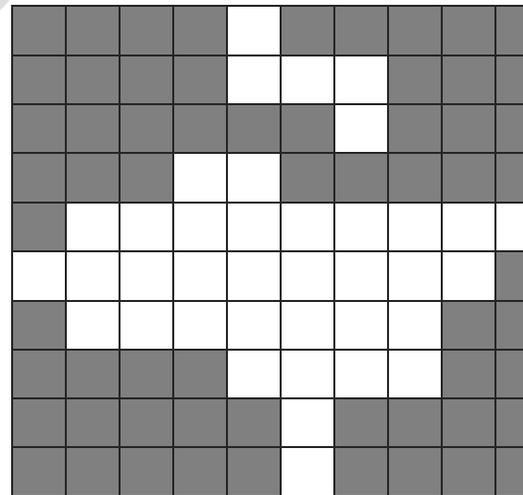
Any two cells (i, j) and (i', j') in X are said to be adjacent if either $i = i'$ and $|j - j'| = 1$ or $|i - i'| = 1$ and $j = j'$.

2.2 Adjacent points of binary image or connected image

Any two cells could be joined through adjacent cells that are connected. If all the points of binary image F are connected then it is called connected image.



Connected binary Image



Not connected Binary Image

Figure 2: Connected binary image and not connected binary image

2.3 Boundary of a binary image

The binary image F says that it contains 1's at some cell (i, j) and 0's at other cells in discrete set X , here after by an image we will refer the cells having value of F as 1 i.e $F(X(i, j)) = f_{ij} = 1$ and the cell or point (i, j) of X will be said to be a point of binary image F , i.e. $f_{ij} = 1 \Rightarrow (i, j) \in F$. The set of all cells (points) (i, j) in F which separate 1's from 0's will be called the boundary of F , thus the boundary of F is represented as

$$B_F = \{(i, j) \in F: \exists \text{ adjacent point } (i', j') \notin F\}$$

2.4 Boundary point

The element $(i, j) \in B_F$ will be called the boundary point. Thus boundary point can be defined alternatively: A point (cell) $(i, j) \in F$ will be called boundary point if at least one of following four properties is satisfied

- (i) $f_{i-1, j} = 0$ or $i = 1$
- (ii) $f_{i, j-1} = 0$ or $j = 1$
- (iii) $f_{i+1, j} = 0$ or $i = m$
- (iv) $f_{i, j+1} = 0$ or $j = n$

Based on this alternative definition boundary points can be categorized in following categories:

If a boundary point satisfies only one property it will be called type 1 boundary point, if any boundary point satisfies any two properties only it will be called type 2 boundary points, if any boundary point satisfies any 3 properties only then it will be called type 3 boundary point. A boundary point satisfying all four properties will be an isolated point of image F and will be called type 4 boundary point.

2.5 Inner point

A cell (point) $(i, j) \in F$ will be the inner point of binary image if it is not the boundary point. Thus inner point is a point $(i, j) \in F$ and $(i, j) \notin B_F$.

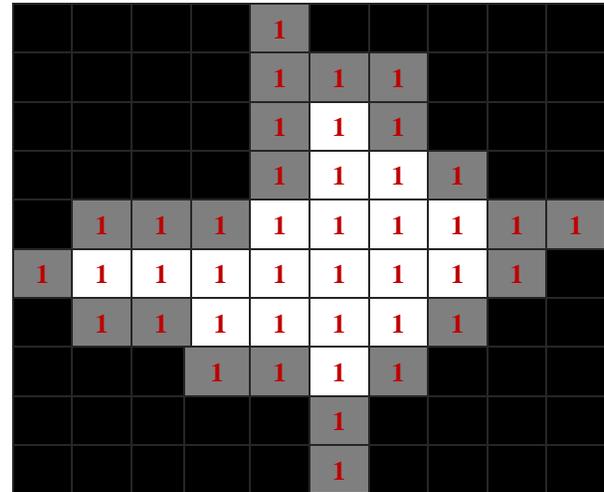


Figure 3: Boundary points and inner points

2.6 Envelop and envelop points of binary image F

The set of all cells (points) in X but not in F , which separate 1's of F from 0's is called the envelop of F .

$$E_F = \{(i', j') \notin F: \exists \text{ adjacent point } (i, j) \in F\}$$

And element $(i', j') \in E_F$ will be called the envelop point. Thus envelop point can be defined alternatively as a point (cell) $(i', j') \notin F$ i.e. $f_{i', j'} = 0$ will be called envelop point if at least one of following four properties is satisfied

- (i) $f_{i-1, j} = 1$
- (ii) $f_{i, j-1} = 1$
- (iii) $f_{i+1, j} = 1$
- (iv) $f_{i, j+1} = 1$

Based on this alternative definition envelop points can be categorized as:

If a envelop point satisfies only one property it will be called type 1 envelop point, if any envelop point satisfies any two properties only it will be called type 2 envelop point, if any envelop point satisfies any 3 properties only then it will be called type 3 envelop point. A envelop point satisfying all four properties will be an hole of image F and will be called type 4 envelop point.

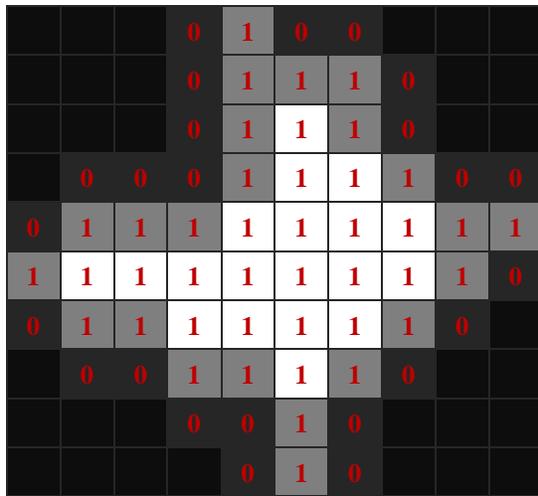


Figure 4: Boundary points, envelope points and inner points

2.7 Boundary point switching

The given projection set has projection in two directions namely horizontal and vertical, so the switching component will contain only four points (cells) in X . If the elements of switching component are adjacent, then it will be called boundary point switching component.

Let the two disjoint subsets I_1 and I_2 of switching component $I_1 \cup I_2$ are $I_1 = \{(i_1, j_1) (i_2, j_2)\}$ and $I_2 = \{(i_1, j_2) (i_2, j_1)\}$

if $I_1 \subset B_F$ ($I_1 \subset E_F$) and $I_2 \subset E_F$ ($I_2 \subset B_F$ respectively) then this switching operation with this kind of switching component is called boundary point switching. The figures explain the definitions.

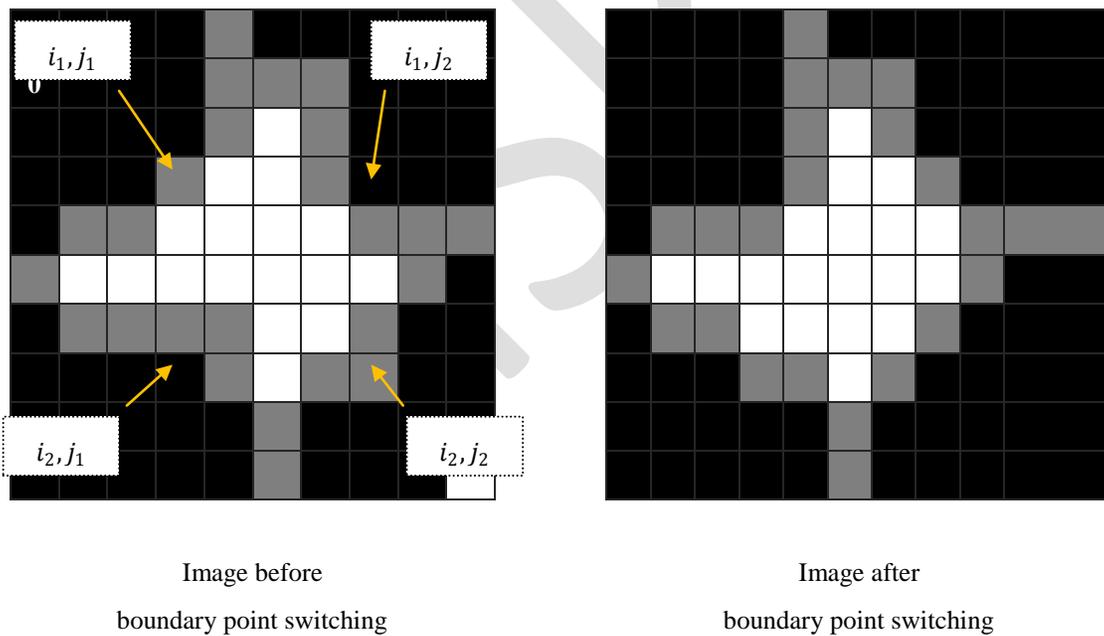


Figure 5: Boundary point switching component and switching result

2.8 Measure of convexity

Let n_h (respectively n_v) denote all horizontally (respectively vertically) adjacent cells in F , n_h (respectively n_v) can be used as a measure of horizontal (respectively vertical) convexity.

Let n_h denote all horizontally adjacent cells in F , also n_h can

be used as a measure of horizontal convexity, similarly, n_v may be considered for vertical convexity denoting all vertical adjacent cells in F .

Thus for the binary image $F = (f_{ij})_{m \times n}$ having projection set $\mathcal{P}(R, C)$ with $R = \{r_1, r_2, \dots, r_m\}$, $C = \{c_1, c_2, \dots, c_n\}$

$$n_h = \sum_{i=1}^m \sum_{j=1}^{n-1} f_{ij} f_{ij+1}$$

and

$$n_v = \sum_{j=1}^n \sum_{i=1}^{m-1} f_{ij} f_{i+1j}$$

Let us define a measure for *hv*-convexity as

$n_{hv} = n_h + n_v$ = number of horizontally and vertically adjacent cells in F , so

$$n_{hv} = \sum_{i=1}^m \sum_{j=1}^{n-1} f_{ij} f_{ij+1} + \sum_{j=1}^n \sum_{i=1}^{m-1} f_{ij} f_{i+1j} \tag{1}$$

it may be noted that

$$n_h \leq \sum_{i=1}^m r_i - m$$

where r_i is the i^{th} horizontal projection set $\mathcal{P}(R,C)$. Since number of adjacent cells in i^{th} row of F will always be less than or equal to the number of 1's in i^{th} row of F . If the image is h-convex then

$$n_h = \sum_{i=1}^m r_i - m$$

Similarly

$$n_v \leq \sum_{j=1}^n c_j - n$$

and if image is v-convex, then

$$n_v = \sum_{j=1}^n c_j - n$$

Thus

$$n_{hv} = n_h + n_v \leq \sum_{i=1}^m r_i - m + \sum_{j=1}^n c_j - n$$

Since the projection set $\mathcal{P}(R,C)$ has to be compatible for reconstruction problem. Thus

$$\sum_{i=1}^m r_i = \sum_{j=1}^n c_j = N = \text{total number of cells in } F.$$

Hence

$$n_{hv} \leq 2 \sum_{i=1}^m r_i - m - n = \sum_{j=1}^n c_j - m - n$$

And if the image is *hv* convex then

$$n_{hv} = 2 \sum_{i=1}^m r_i - m - n = 2 \sum_{j=1}^n c_j - m - n = 2N - (m + n) \tag{2}$$

Thus for reconstruction of *hv* convex images, the fitness (cost) function should be to achieve convexity, thus here the optimality criterion has been considered to maximize

$$n_{hv} = 2 \sum_{i=1}^m r_i - (m + n)$$

III. RECONSTRUCTION ALGORITHM BASED ON SIMULATED ANNEALING

Our reconstruction problem (F_{hv}, \mathcal{P}) with $\mathcal{P} = P(R, C)$ has been transformed to optimization problem

$$\max(n_{hv})$$

Subject to

$$\begin{aligned} \sum_{i=1}^m f_{ij} &= c_j \quad \forall j = 1(1)n \\ \sum_{j=1}^n f_{ij} &= r_i \quad \forall j = 1(1)m \\ \sum_{i=1}^m r_i &= \sum_{j=1}^n c_j \end{aligned}$$

The proposed reconstruction algorithm based on Simulated Annealing approach:

3.1 Initial solution

The initial solution of reconstruction (F_{hv}, \mathcal{P}) is obtained using any basic reconstruction algorithm, which does not provide the *hv* convex image.

3.2 Annealing process

The boundary point switching operation has been considered the free movement of the atoms, so the atoms are boundary

point switching components. At highest temperature, which is starting point, all boundary points switching components have equal probability to be chosen, as the temperature goes down, the cooling is controlled with changing the assignment of probability of boundary point switching components to be chosen for switching operation. The assignments of probabilities to these switching components have been done according to category of boundary points and envelop points taken together in the switching component. In our algorithm it has been assigned higher value to a boundary point switching component, if the possibility of improvement in measure of convexity is higher.

Thus, in a typical iteration the boundary point switching operation is performed on all those switching components which have been assigned highest probability, and at that temperature (iteration) solution is obtained. In next iteration, again the boundary points, envelop point and boundary point switching components in new obtained solution will be identified and another assignment of probability is done according the category of boundary points and envelop points. The new solution is obtained in similar manner as in previous iteration. Thus this solution is at next lower temperature (or at next iteration).

3.3 Termination

The freezing point or final solution is achieved, when we stop the iterations. The stopping criteria has been taken as, when either change in assignment of probabilities to the switching components is not significant (not possible) or the change in assignment of probabilities does provide possibility to change in the measurement of convexity (or in the value of cost/fitness function), whichever is achieved earlier.

IV. IMPLEMENTATION OF RECONSTRUCTION ALGORITHM

The proposed reconstruction algorithm is implemented as

Algorithm

Step 1: Get initial solution F^0 using algorithm 2.7.2 and set

$tm := Maxt$ and $niter:=1$

Step 2: Find B_F the set of boundary points and E_F set of envelope points of F .

Step 3: If $niter > tm$ then exit

Step 4: For each element of B_F find all boundary point switching component and according to their category assign the probability as highest probability to switching component have isolated point or type 4 boundary points. And in order of the type of boundary point assign the probability to all switching components.

Step 5: Perform the switching operation on switching components with maximum probability in F

Step 6: set $niter = niter + 1$. If the value of cost function N_{hv} is improved then Go to step 2 and set $tm = niter + maxt$. Else Go to step 4 and choose another component in B_F with next highest probability to perform switching operation and increase.

Step 7: Repeat steps 2 to 6 until improvement in n_{hv} is significant.

V. EXPERIMENTAL RESULTS

For experimental study the test set of convex binary images of sizes 10x10, 20x20, 30x30, 40x40, 50x50, 60x60, 70x70, 80x80, 90x90 and 100x100 are generated, for each size a set of 100 images has been generated. For each image the horizontal and vertical projections are calculated as row sum and column sum of corresponding binary matrices. Thus a image of size $n \times n$ will have projection set $\mathcal{P}(R, C)$ with $R = \{r_1, r_2, \dots, r_m\}$, $C = \{c_1, c_2, \dots, c_n\}$. A typical example of a binary image of size 20x 20 and its projection set $\mathcal{P}(R, C)$ are given in figure 3.6.



$R = \{ 2, 3, 4, 5, 5, 5, 7, 8, 10, 12, 12, 12, 13, 18, 12, 11, 10, 4, 4, 2,$

$C =$

$\{ 3, 6, 8, 14, 15, 18, 20, 20, 13, 9, 8, 8, 6, 4, 2, 1, 1, 1, 1, 1.$

Figure 6: Image 20x20 and its Projection Set

On these image set we implemented the proposed reconstruction algorithm, we also implemented the algorithm

given in [20] on these images. The performance of these algorithms is compared with measure of convexity defined in section 2.8 and the reconstruction error defined as normalized L_1 error given in following

$$E_R = |F - \tilde{F}| = \frac{100}{mn} \sum_{j=1}^n \sum_{i=1}^m |f_{ij} - \tilde{f}_{ij}| \tag{3}$$

Where F and \tilde{F} are the original image and reconstructed images respectively.

Table 1: Comparison of (n_{hv}) measure of convexity of proposed algorithm with existing algorithm for the typical images

Size of images	Bound	(n_{hv}) measure of convexity		
		Initial guess	Final reconstruction	
			Existing algorithm	Proposed algorithm
10x10	156	155	156	156
20x20	278	265	270	270
30x30	626	611	616	616
40x40	1603	1561	1572	1588
50x50	2678	2646	2655	2657
60x60	3130	3081	3097	3101
70x70	8470	8406	8427	8447
80x80	11058	10989	11014	11034
90x90	15009	13885	13928	13961
100x100	15130	14804	14806	14981

Table 2: Comparison of reconstruction error of proposed algorithm with existing algorithm for the typical images

Size of images	(E_R) Reconstruction error		
	Initial guess	Final reconstruction	
		Existing algorithm	Proposed algorithm
10x10	4.00	0	0
20x20	8.00	6.50	4.50
30x30	3.78	4.40	3.30
40x40	5.25	5.63	6.62
50x50	3.76	4.24	4.00

60x60	3.94	3.78	3.28
70x70	2.04	2.24	2.04
80x80	2.06	2.00	2.06
90x90	2.37	2.32	1.88
100x100	9.04	8.80	8.06

Table 3: Comparison of reconstruction result of proposed algorithm with existing algorithm for the average results

Size of images	Existing algorithm		Proposed algorithm	
	n_{hv}	E_R	n_{hv}	E_R
10x10	138.20	0.00	138.20	0.00
20x20	617.10	0.95	617.60	0.65
30x30	1422.10	2.60	1425.00	2.04
40x40	2567.30	3.09	2574.20	2.41
50x50	4038.20	3.36	4051.40	3.05
60x60	5831.90	3.62	5854.30	3.61
70x70	7928.90	3.36	7954.30	2.98
80x80	10406.30	3.31	10442.80	3.05
90x90	13148.40	3.72	13206.70	3.46
100x100	16313.90	3.70	16378.40	3.34

Table 4: Comparison of n_{hv} and E_R of proposed algorithm with existing algorithm for the average results with 1% noise

Size of images	Existing algorithm		Proposed algorithm	
	n_{hv}	E_R	n_{hv}	E_R
10x10	138.20	0.00	138.20	0.00
20x20	617.10	0.80	617.60	0.65
30x30	1414.10	3.41	1409.80	3.61
40x40	2557.50	3.58	2553.90	3.46
50x50	4029.40	3.67	4012.00	3.77
60x60	5824.80	4.14	5814.40	4.10
70x70	7910.10	3.79	7908.80	3.83
80x80	10382.50	3.75	10360.20	3.72
90x90	13134.50	4.05	13045.50	4.28
100x100	16297.70	3.90	16234.50	4.00

Table 6: Comparison of n_{hv} and E_R of proposed algorithm with existing algorithm for the average results with 3% noise

Size of images	Existing algorithm		Proposed algorithm	
	n_{hv}	E_R	n_{hv}	E_R
10x10	138.20	0.00	138.20	0.00
20x20	617.10	0.95	617.60	0.65
30x30	1421.90	2.71	1425.00	2.04
40x40	2567.10	3.09	2574.20	2.41
50x50	4034.80	3.36	4043.50	3.07
60x60	5803.00	4.20	5768.10	4.23
70x70	8687.79	3.86	7842.60	4.02
80x80	10381.30	3.71	10348.40	3.67
90x90	13135.30	3.99	13124.00	3.85
100x100	16318.00	3.78	16294.20	3.76

Table 5: Comparison of n_{hv} and E_R of proposed algorithm with existing algorithm for the average results with 2% noise

Size of images	Existing algorithm		Proposed algorithm	
	n_{hv}	E_R	n_{hv}	E_R
10x10	138.20	0.00	138.20	0
20x20	609.40	2.68	605.40	2.55
30x30	1408.70	4.17	1407.20	4.10
40x40	2554.70	3.73	2541.40	3.76
50x50	4027.40	3.80	4018.50	3.59
60x60	5799.30	4.55	5792.30	4.56
70x70	7900.60	4.04	7886.10	4.39
80x80	10333.90	4.18	10305.30	4.18
90x90	13093.90	4.27	13046.90	4.43
100x100	16251.70	4.21	16146.70	4.39

Table 7: Comparison of n_{hv} and E_R of proposed algorithm with existing algorithm for the average results with 4% noise

Size of images	Existing algorithm		Proposed algorithm	
	n_{hv}	E_R	n_{hv}	E_R
10x10	138.20	0.00	138.20	0.00
20x20	604.30	3.98	596.50	4.38
30x30	1401.60	4.86	1392.40	4.96
40x40	2542.90	3.96	2525.70	4.38
50x50	4004.50	4.50	3991.60	4.18
60x60	5783.90	4.93	5766.70	4.68
70x70	7855.10	4.61	7762.30	4.93
80x80	10316.30	4.50	10267.50	4.68
90x90	13015.60	4.78	12955.90	4.92
100x100	16165.00	4.70	16107.00	4.77

algorithm with existing algorithm for the average results with 5% noise

Size of images	Existing algorithm		Proposed algorithm	
	n_{hv}	E_R	n_{hv}	E_R
10x10	138.20	0.00	138.20	0.00
20x20	607.60	3.78	594.50	5.18
30x30	1402.60	4.96	1392.80	5.18
40x40	2528.20	5.28	2506.80	5.28
50x50	3994.10	5.00	3979.90	4.82
60x60	5757.00	5.26	5699.20	5.36
70x70	7826.50	4.80	7798.00	4.97
80x80	10281.20	4.85	10115.30	5.25
90x90	13011.10	4.99	12942.20	5.25
100x100	16137.70	4.98	16055.30	5.06

Table 8: Comparison of n_{hv} and E_R of proposed

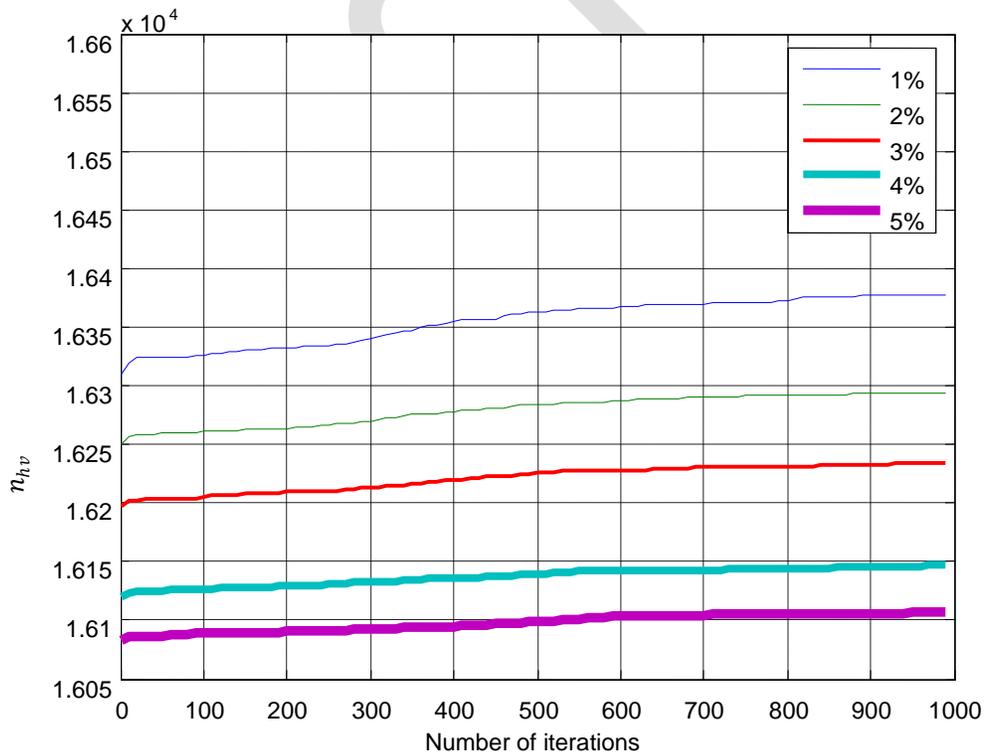


Figure 7: n_{hv} with noisy projections for 40x40 image

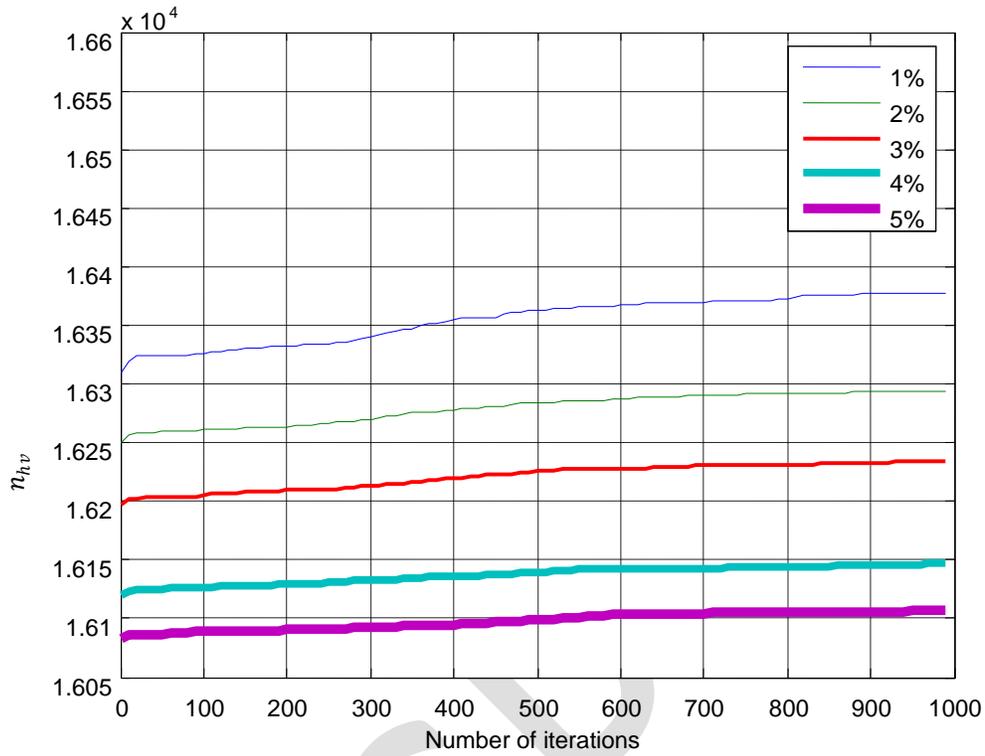


Figure 8 n_{hv} with noisy projections for 100x100 images

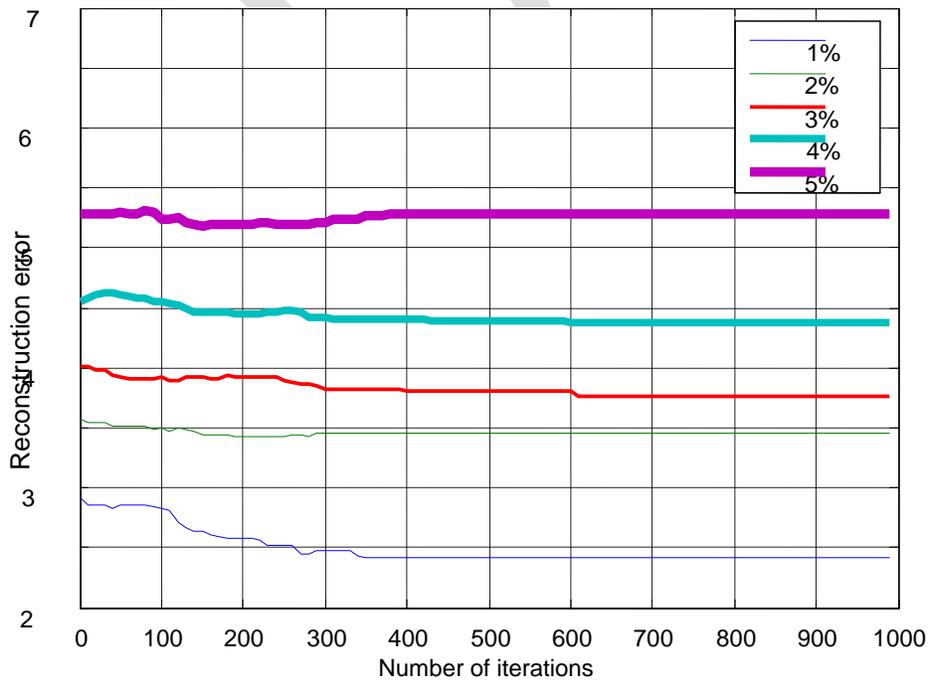


Figure 9: Reconstruction error with noisy projections for 40x40 images

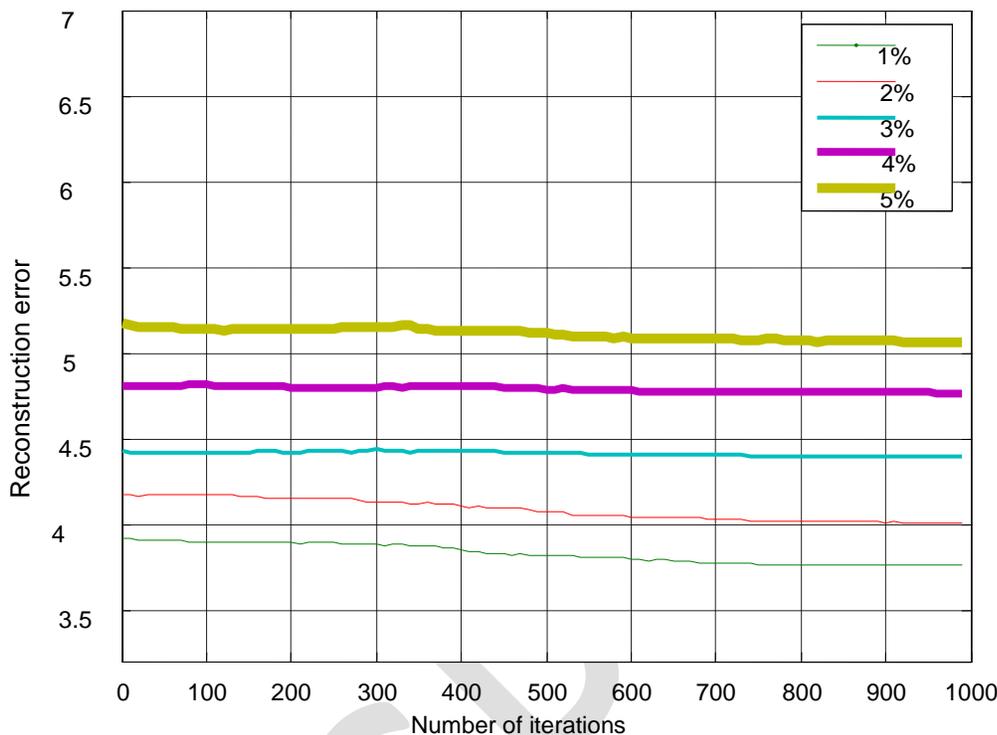


Figure 10: Reconstruction error with noisy projections for 100x100 images

V. CONCLUSION

Finally, it is evident that proposed algorithm converges faster than existing algorithm [20], it also gives good visual reconstruction as evident from figures. In case of noisy projections up to 2% of noise, our algorithm gives better reconstructions than existing algorithm [20] but with higher noise the reconstructions are not that good, the reason for this may be that we are using only boundary of images which get more distorted with higher level of noise.

REFERENCES

[1] Kak, A.C., Slaney, M.: Principles of Computerized Tomographic Imaging, IEEE Press New York (1988)

[2] Hadamard, Jacques: Lectures on Cauchy's Problem in Linear Partial Differential Equations, Dover Publications (1923)

[3] Alberto Del Lungo, Andrea Frosini, Maurice Nivat and Laurent Vuillon: Discrete Tomography: Reconstruction under

Periodicity Constraints, Lecture Notes in Computer Science (2002)

[4] Birgit van dalen:Boundary length of reconstruction in discrete tomography(2010)

[5] Fethi jarray,marie-Christine costa christophe Picouleau:Approximating hv-convex binary matrix and images from discrete projection LNCS pp 413-422, Springer-Verlag Berlin Heidelberg (2008)

[6] Fethi jarray,Ghassen Tlig: A simulated annealing for reconstructing hv-convex binary matrix pp 447-454, Electronic notes in discrete mathematics 33(2010)

[7] Barucci, E., Del Lungo, A., Nivat, M. and Pinzani, R. Reconstructing convex polyominoes from their horizontal and vertical projections, Theoret. Comput. Sci. 155 321–347. (1996).

[8] Brunetti, S. and Daurat, A. An algorithm reconstructing convex lattice sets, Theoret. Comput. Sci. (Issues 1–3) 304 (28) 35–57. (2003).

- [9] Chrobak, M. and Dürr, C. Reconstructing hv-convex polyominoes from orthogonal projections Inform. Process. Lett. 69 283–289.(1999).
- [10] Ryser, H.J. Combinatorial properties of matrices of zero and ones. Can. J math9, 371-377(1957).
- [11] Gale, D. A theorem on flows in networks. Pacific J. Math 7, 1073-1082 (1957).
- [12] Chang, S.K The reconstruction of binary patterns from their projections. Commun. ACM 14, 21–25.(1971).
- [13] Kuba, A. and Herman G.TDiscrete Tomography: Foundations, Algorithms, and Applications, Birkhauser. (1999).
- [14] Balazs, P., On the ambiguity of reconstructing hv-convex binary matrices with decomposable configurations, Acta Cybernetica, vol.18, pp. 367–377, (2008).
- [15] Del Lung, A., Nivat, M., and Pimania, R., The number of convex polyominoes reconstructible from their orthogonal projections, Discrete. Mathematics, vol. 157, pp. 65-78, (1996).
- [16] Kirkpatrick, S., Gelatt, C.D. and Vecchi, M. P., Optimization by Simulated Annealing, Science, New Series, vol. 220, No. 4598, pp.671-680, (1983).
- [17] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E., Equation of State Calculations by Fast Computing Machines. The Journal of Chemical Physics, vol. 21 (6), pp.1087-1092, (1953).
- [18] Cerny, V., Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. Journal of Optimization Theory and Applications, vol.45, pp.41–51, (1985).
- [19] Jarray, F., Costa, M.C. and Picouleau, C., Approximating hv-Convex Binary Matrices and Images from Discrete Projections, DGCI 2008, LNCS, vol. 4992, pp.413–422, (2008).
- [20] Jarray, F., Tlig, G., A simulated annealing for reconstructing hv-convex binary matrix ,Electronic notes in discrete mathematics, vol. 33, pp 447-454, (2010).