Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

The Application of Simulated Annealing to a Map Seeking Circuit

Sean Martin, Pedro Rodriguez, Patricia Murphy
The Johns Hopkins University Applied Physics Laboratory
Laurel, USA
Sean.Martin@jhuapl.edu, Pedro.Rodriguez@jhuapl.edu, Patricia.Murphy@jhuapl.edu

***Abstract*—This paper examines how to apply simulated annealing to a map seeking circuit (MSC) when the circuit is formulated as a gradient ascent optimization problem. This modification is performed on a map seeking circuit solving the three dimensional model correspondence problem. It is shown that when the input image is partially occluded and covered by random edges, then the MSC can converge incorrectly to a local maximum instead of the global maximum. Under these conditions, simulated annealing is shown to improve the performance of the MSC through a series of experiments involving the Hausdorff distance metric.**

***Index Terms*—Simulated Annealing, Map Seeking Circuit**

I. Introduction

The map seeking circuit (MSC) is a relatively new algorithm designed to reduce the computational load while attempting to solve a transformation-discovery problem where the objective is to find the best transformation that maps a reference pattern to a target pattern. The MSC is biologically inspired and has found applications to machine vision related problems [2] [3] [1]. The algorithm operates by superimposing a sparse model image subjected to various transformations (rotation, scaling, etc.) and comparing this superimposition with another superimposition of the input image subjected to the inverse transformations. By working in this superimposition space, the MSC is able to reduce the dimensionality of the search problem while finding the best transforms to match a model to a given input image. This search space dimensionality is reduced from the product of all possible transforms to their sum [1]. However, sparse input images are required for correct convergence. To create sparse images, typically the images are preprocessed with an edge detector. Thus only the edges in an image are superimposed.

There are strong algorithms in the literature for finding matches between images. For example, variations on Lowe's SIFT algorithm have been proven to be robust in the presence of occlusion [11]. However, this algorithm suffers from issues relating to three dimensional projections and rotations when attempting to find a match. For such problems, the MSC has been shown effective [1].

The MSC is generally applied to constrained maximization problems associated with correspondence maximization [9]. This paper discusses the model correspondence problem which involves mapping the projection of a single transformed three dimensional model into a two dimensional image, and then using the MSC to determine those transforms that best match the input image. This becomes more challenging when the input image is obscured by noise or occlusion. As multiple researchers have pointed out, imperfections from occlusion in the input image results in a multimodal objective function and can cause the circuit to converge to local maxima vice global maxima which becomes the focus of this work [5] [3].

Many techniques are available for optimizing multimodal functions. Stochastic optimization has been particularly effective with addressing such functions by randomly perturbing the steps chosen during optimization. Some examples include: probabilistic tabu search, particle swarm optimization (PSO), evolutionary algorithms (EA), and simulated annealing (SA) [7] [4] [14]. However, algorithmic variations on many of these approaches tend to be population based in the sense that they compare candidate solutions together based off a single function. As shown in section II-A, the MSC is composed of recursive functions that perform separate optimizations for their respective transformation layers. Further, the MSC begins as nearly a constant function since the gains are all initialized to values of 1. In addition, the MSC uses gradient information during incremental updates which suggests any enhancements should be compatible with gradient based optimizers. It follows that the use of a simulated annealing schedule to control the amount of random perturbation during gradient processing would fit well with the design of the MSC. Since simulated annealing is provably convergent to a global maximum given the right annealing schedule [12], a strong argument can be made that an annealing MSC will retain the MSC's convergent properties in the presence of noisy inputs.

II. Background

A. The Map Seeking Circuit

The MSC attempts to find the best transformation T mapping a given input image I to a model M. This transformation typically covers several classes of transforms (i.e. scaling, translation, rotation). Each of these transform classes is represented by a separate layer L such that the full transformation is a composition of L maps.

$$T = T_{i_L}^{(L)} \circ \cdots \circ T_{i_2}^{(2)} \circ T_{i_1}^{(1)} \tag{1}$$

where i_L denotes a specific transform i of layer L (i.e. 10 degrees of rotation). Following the notation in [3], the inner product between the transformed input and the model can thus be written as:

$$c(T) =< T(I), M > \tag{2}$$

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

More specifically, the correspondence function the MSC seeks to maximize is defined as:

$$f(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(L)}) := \sum_{i_1=1}^{n_1} \cdots \sum_{i_L=1}^{n_L} < T_{i_L}^{(L)} \circ \cdots \circ T_{i_2}^{(2)} \circ T_{i_1}^{(1)}(I), M > x_{i_1}^{(1)} \cdots x_{i_L}^{(L)} \quad (3)$$

where $\mathbf{x}^{(L)}$ is the collection of all gain coefficients in a given layer and $x_{i_L}^{(L)}$ is the gain for a specific transform i in layer L. Between layers in the MSC, each transform $T_{i_L}^{(L)}$ is applied to the layer input, scaled by the respective gain $x_{i_L}^{(L)}$, and added to all the other gain scaled transforms of that layer to form a superposition which is passed off to the next layer.

The map seeking circuit algorithm seeks to maximize the function f by iterating each layer L in order and updating that layer's gain coefficients $\mathbf{x}^{(L)}$. This is computed through the use of a competition function C:

$$\mathbf{x}(n+1) = C(\mathbf{x}(n), < T(I), M >) \quad (4)$$

There are multiple ways of defining C, for this paper the following is considered:

$$C_i^l(x_i, v_i) := \begin{cases} max(0, x_i - k^l(1 - \frac{v_i}{max(v)})) & if\ max(v) \geq \varepsilon^l \\ 0 & if\ max(v) < \varepsilon^l \end{cases} \quad (5)$$

where v_i is the dot product for transform i on layer l and $max(v)$ represents the highest dot product value for a particular layer. Both ε and k are constants.

The circuit in this paper contains two layers consisting of 3D rotation (i.e. azimuth and elevation) and image plane translation (i.e. columns and rows). Specific parameters are discussed in section IV. The design is nearly identical to figures 2-5 and 2-6 in Arathon's text with the memory feeding into the back layer and the image input feeding into the front layer [1].

B. Simulated Annealing

The basic simulated annealing algorithm, as described in [14], allows a gradient based optimizer to move in a suboptimal candidate direction based on the distance D between the objective function value of the candidate direction and the gradient direction at the state of the current optimizer iteration. This happens whenever the following condition is met:

$$e^{-\triangle D/t_m} > R(0,1) \quad (6)$$

Where t_m is the temperature on iteration m and R is a uniform random distribution between 0 and 1.

Unlike some stochastic optimizers, simulated annealing can be proven to converge to a global optimum given an annealing schedule designed for the optimization problem's search space [12] [8]. The proof of convergence depends on discrete Markov chain analysis over search space states visited during each step of the optimization algorithm. This is modeled by a state probability vector which is propogated with a transition model at each iteration of the optimization. To be more specific, the state probability vector x at iteration m is defined as:

$$x(m) \triangleq [x_1(m), x_2(m), \ldots, x_j(m)] \quad (7)$$

for a search space containing j states, and calculated as:

$$x(m) = x(0) \prod_{j=0}^{m-1} P(t_m) \quad (8)$$

where $P(t_m)$ is the state transition matrix at iteration m:

$$P_{ij}(t_m) = \begin{cases} 0 & j \notin N(i) \\ \frac{1}{|N(i)|} \min\{1, e^{-\triangle D/t_m}\} & j \in N(i) \end{cases} \quad (9)$$

based on the annealing schedule t_m, neighbors $N(i)$ of state i, and the distance D from equation 6.

Given these definitions, and a properly constructed annealing schedule t_m, it is then possible to prove that the underlying Markov chain model is strongly ergodic [12]. In other words, the simulated annealing algorithm will convege from any initial starting state to a global maximum q:

$$\lim_{m \to \infty} \sup_{x(0)} \|x(m) - q\| = 0 \quad (10)$$

C. Hausdorff Metric

Given point sets $A = \{a_1, \ldots, a_p\}$ and $B = \{b_1, \ldots, b_p\}$ which are both finite, the Hausdorff distance is defined as

$$H(A,B) = max(h(A,B), h(B,A)) \quad (11)$$

where

$$h(A,B) = \max_{a \in A} \min_{b \in B} \|a - b\| \quad (12)$$

and for this paper $\|\cdot\|$ refers to the L_2 norm, although any norm is applicable. Intuitively, the Hausdorff distance measures the amount of mismatch between two sets by measuring the furthest point in A from any point in B and vice versa [10].

III. An Annealing Map Seeking Circuit

Gedeon and Arathorn characterize the MSC process in the correspondence problem mathematically in their work on analyzing convergence [3]. Specifically, in their equation (8) they model the problem with an objective function relating a stored gain on each possible transform for each layer to the product of the gain and the dot product between the forward and backward inputs to the circuit on each layer. Writing the equations from section II-A more succinctly for a single layer l:

$$f(x_i^l) = x_i^l < T_i^l(I^{l-1}), M^l > \quad (13)$$

Where x_i is the gain for each transform T_i on the layer l memory of superimposed inverse model transforms M^l and input image I^{l-1} coming from the previous layer. They note that the gradient w.r.t. the gain of this function is trivially the dot product and it follows that the competition function in their equation (10) is a form of gradient ascent. It also follows that when local maxima exist during the convergence process the circuit may not reach the global maximum as required. This is shown empirically in section IV.

For the work in this paper, D is calculated from equation (13) as the difference between $f(x_i)$ and $f(x_{max(v)})$ where

$max(v)$ represents the highest dot product value for a particular layer. Exact values for t_m are provided in section IV. The competition function from equation 10 in [3] for each circuit layer l is modified as follows to account for when the condition from equation (6) is met:

$$C_i^l(x_i, v_i) := \begin{cases} x_i & if\ e^{-\triangle D/t_m} > R(0,1) \\ & else \\ max(0, x_i - k^l(1 - \frac{v_i}{max(v)})) & if\ max(v) \geq \varepsilon^l \\ 0 & if\ max(v) < \varepsilon^l \end{cases} \quad (14)$$

Where both ε and k are constants. The gain value returned by this competition function is then assigned to x_i. For the experiments in this paper, ε is set to 0 to prevent premature convergence and k is set to 50 in the rotation layer and 2 in the translation layer. Since ε is set to 0 no gains are ever zeroed out and this can be seen in the Hausdorff metric plots never reaching a value of 0 in figures 3 and 5. The addition of simulated annealing in the competition function prevents the circuit from strictly taking the best value at each layer and allows other layers to converge slower to prevent convergence to a local maximum.

While a formal proof of convergence to the global maximum for an annealing MSC requires significant analysis [3] [5], it is worth noting that the annealing MSC can be reformulated into the same Markov chain analysis performed to prove the convergence of simulated annealing. The difference is that the annealing MSC runs a competition function over each layer separately and thus contains a separate Markov chain for each layer of the circuit. Thus, the Markov chain state and state transition calculations from section II-B can be redefined as follows in terms of the gains x on layer l containing a set of transformations $\{1, \ldots, n_l\}$:

$$x^l(m) = [x_1^l(m), x_2^l(m), \ldots, x_{n_l}^l(m)] \quad (15)$$

$$P_{ij}(t_m) = \frac{1}{n_l} \min\{1, e^{-\triangle D/t_m}\} \quad j \in \{1, \ldots, n_l\} \quad (16)$$

Note that D depends on equation 13 and thus depends on other layers of the circuit modeled in separate Markov chains. Also note that the construction of a convergent annealing schedule will additionally depend on the input image as seen in the following experimental section when the amount of edges and occlusion are increased. For these reasons the experiments in the following section do not follow a provably convergent annealing schedule. Analysis on such a schedule is left to future work.

IV. Experiments

This section looks at scenarios that have the potential to cause the circuit to converge to a local maximum. All images in this section are 400x400 pixels. The rotation layer consists of 12 rotations over the x and y axis at 30 degrees each. The translation layer consists of 100 image shifts at 40 pixels each in the x and y direction. The image wraps around the screen due to the circular shift function implemented. To avoid perspective projection issues, all projections are orthographic. To ensure consistency, for all randomly generated noise, the same noisy input image is used in all experiments (i.e. the noise is not regenerated on each run). The model employed in all experiments is a truck model from the Princeton Shape Benchmark repository [15]. This model is used exclusively in this paper. To simulate edge detection, a wireframe model is employed with hidden line removal to simulate only visible edges. This is shown in figure 1. In each experiment the model is transformed by a 100 pixel translational offset in the x and y axes, 60 degree x axis rotation, and 150 degree y axis rotation. The resulting orthographic projection is perturbed with various types of noise and then fed into the circuit as an input image. The goal is to determine the correct model transformations from the input image. In each experiment the circuit outputs are displayed at the iteration when the circuit either converges or diverges, and half way through the iterations leading to convergence/divergence. For both experiments the initial circuit outputs were initially identical since the gains all begin at a value of 1.0 and then gradually decrease as the circuit iterates. The implementation used in this section is written in C++ and employs openGL [13] to perform all image and model manipulations along with the FFTW fast Fourier transform [6] to conduct dot products over the translational transform layer by performing a convolution.

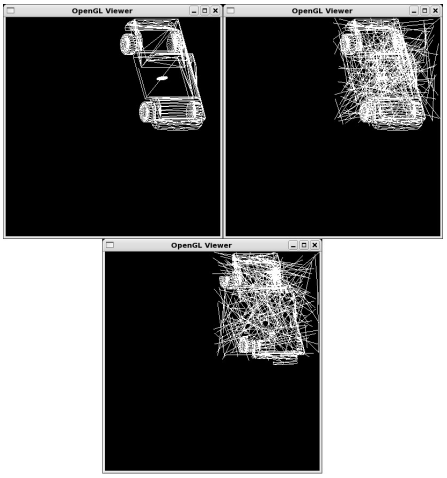

Fig. 1. Input images created through transformed versions of a three dimensional wireframe truck model. Clockwise from upper left: the transformed model with no edges or occlusion, edges applied to the transformed model, edges and occlusion applied to the transformed model.

Since the circuit starts out with all gains set to 1, the temperature schedule begins cold before turning hot and then annealing back to cold. The exact schedule is as follows where m represents the MSC iteration:

$$t_m = \begin{cases} 0 & m < 15 \\ .25 & 15 < m < 90 \\ .1 & 90 < m < 100 \\ 0 & 100 < m \end{cases} \quad (17)$$

Note that simulated annealing is a stochastic process and thus does not always provide convergence to a global max-

imum unless a provably convergent annealing schedule is used. Further note that this temperature schedule extends the circuit iterations which will require more computation. A more ambitious schedule could be used that uses less iterations, however, the methodology of determining an exact schedule is beyond the scope of this paper. This paper only seeks to answer whether or not simulated annealing can improve the performance of the MSC. The result of a successful run without simulated annealing is shown below in figure 2. An unsuccessful run where the MSC converges to a local maximum is shown in figure 4 Note that the two left side images on the bottom row of images in figure 4 differ from those in figure 2. To confirm that the true global maximum in each of these experiments is the same, even in the presence of noise and occlusion, an exhaustive search was conducted over all possible transforms. The inner product was maximized at the model transforms corresponding to the input image transforms.

To see how well the approach works over several MSC runs using simulated annealing, several experiments were performed to assess how the modified circuit will perform on average. Each experiment involved performing 100 runs of the circuit with simulated annealing iterating 110 times. This was compared against single runs of the deterministic MSC algorithm without simulated annealing. The resulting probability that the circuits coverged to the correct transforms along with the Hausdorff distances were recorded and plotted against increasing amounts of random edges added around the transformed model. Edges were placed only around the model to maintain a sparse superposition. Otherwise, the MSC will not converge [3].

A. Input with Random Edges

This section considers a series of MSC runs over the transformed model with random edges applied as shown in figure 1. The number of random edges added were: 130, 175, 220, 265, 310, 355, and 400. The start and end point of these straight line edges were chosen randomly over the upper right quadrant of the input image. The same random edges were applied over each of the 100 MSC runs over all experiments.

Note from figure 2 that the unmodified MSC converges correctly when compared to the transformed model in figure 1. Further note from figure 3 that there is no improvement when simulated annealing is applied in this case, and that for the most part the MSC with simulated annealing performs worse than the regular MSC.

B. Input with Random Edges and Occlusion

This section considers a series of MSC runs over the transformed model with random edges and occlusion applied as shown in figure 1. the same random edges were applied as in section IV-A. Additionally, 75 black squares of dimension 40 pixels were randomly place over the model prior to adding the random edges. This creates the opportunity to have local maximums during the maximization process because an exact match between the transformed input and the model is no longer possible. Now, only part of the model will successfully match the input which will reduce the dot product value and affect the competition function.

Note from figure 4 that the unmodified MSC does not converge correctly when compared to the transformed model in figure 1. Also note that in figure 5 the MSC with simulated annealing outperforms the regular MSC, although performance

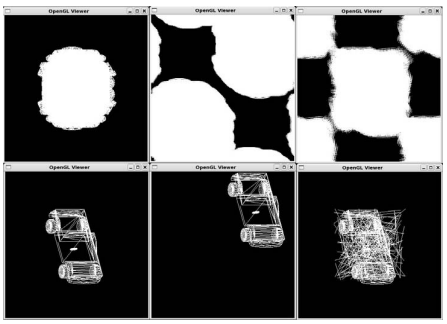

Fig. 2. From left to right: the backward rotation layer output, the backward translation layer output, and the forward translation layer output over an input image containing random edges. The top layer images are after 31 iterations of the circuit, the bottom layer shows 62 circuit iterations.

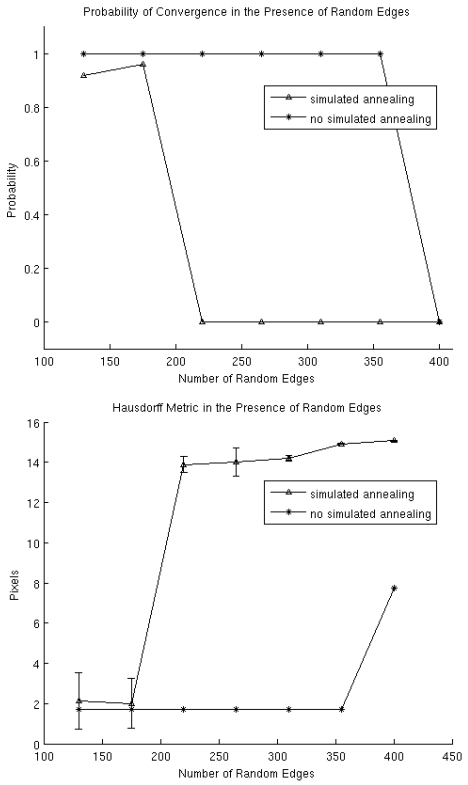

Fig. 3. Plots showing the probability of convergence to the global maximum (top) and the average Hausdorff metric distances with standard deviations (bottom) over 100 MSC runs for different amounts of random edges added to a non-occluded input image.

worsens for both algorithms with the addition of more edges.

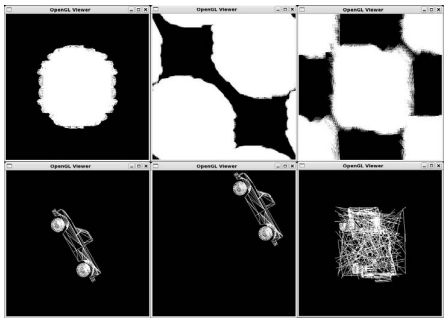

Fig. 4. From left to right: the backward rotation layer output, the backward translation layer output, and the forward translation layer output over an input image containing random edges and occlusion. The top layer images are after 27 iterations of the circuit, the bottom layer shows 54 circuit iterations.

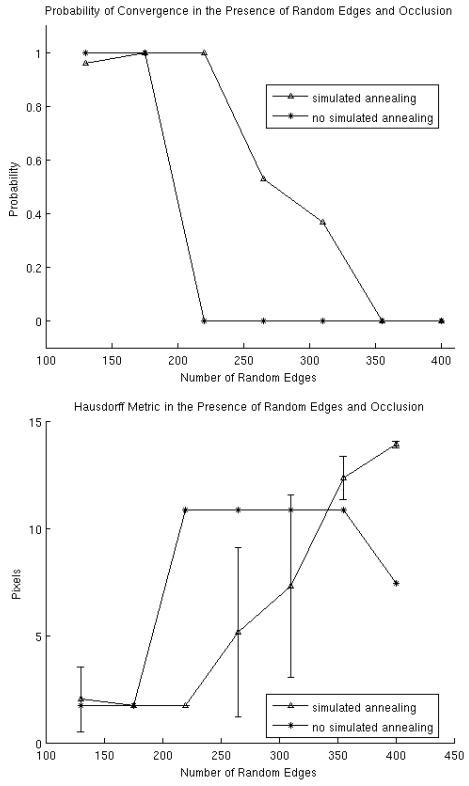

Fig. 5. Plots showing the probability of convergence to the global maximum (top) and the average Hausdorff metric distances with standard deviations (bottom) over 100 MSC runs for different amounts of random edges added to an occluded input image.

C. Analysis of Results

To further analyze the experimental results, first note the sequence of 100 trials follows a binomial distribution where the circuit either does or does not converge correctly to the global maximum on each run. To determine if the runs using simulated annealing differ significantly from the runs using just a MSC, consider a statistical z test over the experiments with 220, 265, and 310 random edges added to the occluded input. Using binomial point estimators with $n = 100$ samples, the z value is calculated as:

$$z = \frac{\hat{p} - p_0}{\sigma_{\hat{p}}} = \frac{Y/n - p_0}{\sqrt{\frac{p_0(1-p_0)}{n}}} \tag{18}$$

In this equation p_0 corresponds to the hypothesis that the MSC and annealing MSC both converge with 99% probability during the experiment with random edges in the input image, and with 1% probability in the experiment with both random edges and occlusion. $\hat{p}$ corresponds either to the to the observed annealing MSC convergence probability shown in figure 3 or figure 5. Since $P(Z > 2.33) = .01$, any z value over 2.33 suggests the hypothesis should be thrown out with 99% confidence. Thus, rejecting the hypothesis in the experiment with only random edges suggests the MSC outperforms the annealing MSC, and rejecting the hypothesis in the experiment with both edges and occlusions suggests that the annealing MSC outperforms the MSC both with high confidence.

When only random edges are added to the input image, $z = 99.50$ for 220, 265, and 310 random edges leading to the conclusion that the regular MSC does outperform the annealing MSC with high confidence. However, when both random edges and occlusion are added to the input image for 220 edges $z = 99.50$, at 265 edges $z = 52.26$, and at 310 edges $z = 36.18$. Thus, it can be concluded that for these experiments the annealing MSC outperforms the regular MSC with high confidence.

V. Conclusion

The work in this paper involved the identification of a specific case where a map seeking circuit does not converge to a correct set of transformations due to random edges and occlusions in the input image. While the annealing MSC does outperform the regular MSC in certain cases when both random edges and large areas of occlusion are present, this result was dependant on a number of variables including the image resolution, the annealing temperature schedule implemented, chosen values for constants, the number of polygons contained in the model of interest, and both the type and amount of noise in the input image. Additionally, the annealing MSC performed worse than the regular MSC when only random edges were present in the input without occlusion.

Regardless, since simulated annealing provably converges to a global maximum [12], and for inputs without noise or occlusion the MSC will either converge to a global maximum or not return a solution [3], the analysis in section III suggests that convergence of the annealing MSC depends on the selection of the annealing schedule. Further work is needed to determine a provably convergent annealing schedule and to reduce the length of schedules requiring many iterations of the algorithm.

ACKNOWLEDGMENT

The authors would like to thank the Precision Engagement Business Area at the Johns Hopkins University Applied Physics Laboratory for sponsorship of this research, and Dr. James Spall for input on an earlier version of this work.

REFERENCES

[1] D. Arathorn, *Map-Seeking Circuits in Visual Cognition: A Computational Mechanism for Biological and Machine Vision*, Stanford, CA: Stanford University Press, 2002.

[2] D. Arathorn, *Computation in the Higher Visual Cortices: Map-Seeking Circuit Theory and Application to Machine Vision*, IEEE AIPR, 2004.

[3] D. Arathorn, T. Gedeon, *Convergence of Map Seeking Circuits*, Journal of Mathematical Imaging and Vision, 29 (2007), pp. 235-248.

[4] R. Eberhart and J. Kennedy, *Particle Swarm Optimization*, IEEE International Conference on Neural Networks, 1995.

[5] Z. Engin, J. Ng, M. Barahona, A. Bharath, *An Annalysis of the Map Seeking Circuit and Monte Carlo Extensions*, Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 2009.

[6] M. Frigo and S. Johnson, *The Design and Implementation of FFTW3*, Proceedings of the IEEE 93 (2), 216-231 (2005). Invited paper, Special Issue on Program Generation, Optimization, and Platform Adaptation. http://www.fftw.org/

[7] F. Glover, *Tabu Search - Part I*, ORSA Journal on Computing, Vol. 1, No. 3, 1989.

[8] B. Hajek, *Cooling Schedules for Optimal Annealing*, Mathematics of Operations Research, vol. 13, pp. 311-329. 1988.

[9] S. Harker, C. Vogel, T. Gedeon, *Analysis of Constrained Optimization Variants of the Map-Seeking Circuit Algorithm*, Journal of Mathematical Imaging and Vision, Vol. 29 , No. 1, 2007.

[10] D. Huttenlocher, G. Klanderman, and W. Rucklidge, *Comparing Images Using the Hausdorff Distance*, IEEE Trans. Pattern Analysis and Machine Intelligence. Vol. 15 No. 9, 1993.

[11] D. Lowe, *Distinctive image features from scale-invariant keypoints*, International Journal of Computer Vision, 60, 2 (2004), pp. 91-110.

[12] D. Mitra, F. Romeo, A. Sanglovanni-Vincentelli, *Convergence and Finite-Time Behavior of Simulated Annealing*, Proceedings of the IEEE Conference on Decision and Control, 1985.

[13] *The openGL Standard Group*, http://www.opengl.org, 2009.

[14] R. Otten, L. van Ginneken, *The Annealing Algorithm*, Boston, MA, Kluwer, 1989.

[15] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, *The Princeton Shape Benchmark*, http://shape.cs.princeton.edu/benchmark/ IEEE International Conference on Shape Modeling and Applications, Genova, Italy, 2004.