

# A Hybrid Genetic Approach for Stereo Matching

Eliyahu Kiperwasser  
Department of Computer  
Science  
Bar-Ilan University  
Ramat-Gan 52900, Israel  
elikip@gmail.com

Omid David\*  
Department of Computer  
Science  
Bar-Ilan University  
Ramat-Gan 52900, Israel  
mail@omidavid.com

Nathan S. Netanyahu†  
Department of Computer  
Science  
Bar-Ilan University  
Ramat-Gan 52900, Israel  
nathan@cs.biu.ac.il

## ABSTRACT

In this paper we present a genetic algorithm (GA)-based approach for the stereo matching problem. More precisely, the approach presented is a combination of a simple dynamic programming algorithm, commonly used for stereo matching, with a practical GA-based optimization scheme. The performance of our scheme was evaluated on standard test data of the Middlebury benchmark [1]. Specifically, the number of incorrect disparities on these data decreases by approximately 20% in comparison to the original approach (without the use of a GA).

## Categories and Subject Descriptors

I.2.10 [Artificial Intelligence]: Vision and Scene Understanding—3D/stereo scene analysis.

## Keywords

Computer Vision, Genetic Algorithm, Stereo Matching

## 1. INTRODUCTION

Stereo vision has been extensively studied by the computer vision community for the last three decades. Robust stereo vision algorithms have many applications, including vehicle/robot obstacle detection, aerial topography acquisition, virtual reality, graffiti detection, and many more.

Stereo vision data are acquired by two (or more) cameras with separate perspectives and are used to determine the depth of an object or a particular point in a scene. The depth information is computed by applying the triangulation principle [19] to the position differences between corresponding points in the (two) images. The latter is referred

\*Web: <http://www.omiddavid.com>

†Nathan Netanyahu is also affiliated with the Center for Automation Research, University of Maryland, College Park, MD 20742 (e-mail: [nathan@cfar.umd.edu](mailto:nathan@cfar.umd.edu)).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO '13, July 6-10, 2013, Amsterdam, The Netherlands.  
Copyright 2013 ACM TBA ...\$15.00.

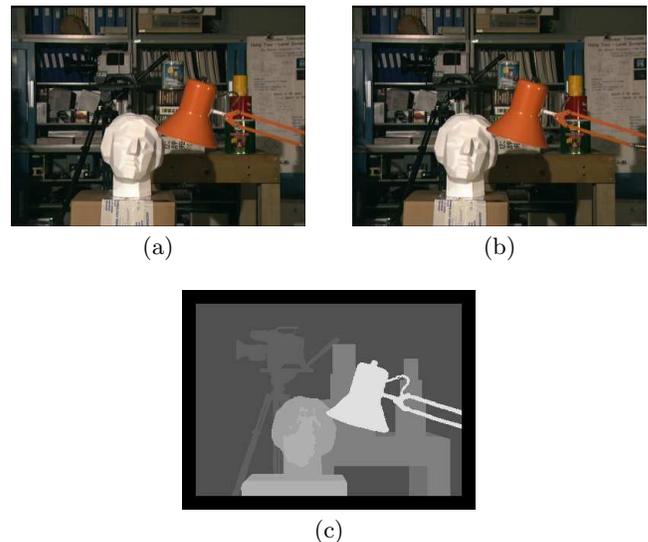


Figure 1: Stereo matching example: (a) The image taken by the left camera, (b) the image taken by the right camera, and (c) the ground truth disparity map.

to as the *image disparity* (at a pixel in question). Before determining the degree of disparity, the most difficult and time-consuming task is to identify corresponding points in the two images. This is the so-called *stereo matching* problem.

The input images to the stereo matching algorithm are assumed to be rectified, i.e., corresponding pixels must lie on the same horizontal line in both images. The output is called the disparity map (see Figure 1), where each pixel contains a displacement value, called the disparity and measured in pixels (from its location in the reference image to its corresponding location in the target image). The maximum distance between corresponding pixels, called the disparity range, is assumed to be known. Pixels in the target image that are occluded will be assigned an interpolated value or some special value encoding their occluded state.

Many algorithms for the stereo correspondence problem have been developed over time. This paper presents a novel evolutionary “boosting” scheme that draws on the well-known scan-line algorithm [14] for the stereo matching problem. We demonstrate that the use of our approach results in approx-

imately 20% increase in accuracy, compared to the performance of the scan-line algorithm without a GA. In principle, the framework can be used to boost other algorithms as well to enhance their performance.

The paper is organized as follows. In Section 2 we survey past work related to the problem of stereo matching. In Section 3 we present in detail our proposed hybrid approach, including the specification of various GA operators, and in Section 4 we present our empirical results. Section 5 makes concluding remarks.

## 2. PREVIOUS WORK

The most basic approach to the stereo correspondence problem compares each pixel from the reference image to its possible matches in the target image and takes the one with the minimum color difference as a match (see Figure 2). Since the input is assumed to be rectified, the only possible matches lie on the same horizontal line within a limited range of pixels specified a priori as the disparity range. This trivial approach does not perform well, as it is very error prone in dealing with smooth surfaces for which all color differences are roughly the same. Furthermore, when taking a picture from two different angles, the color of a certain pixel can change due to lighting conditions, a transparent object in the scene, etc. Such problems render this approach impractical for most cases.

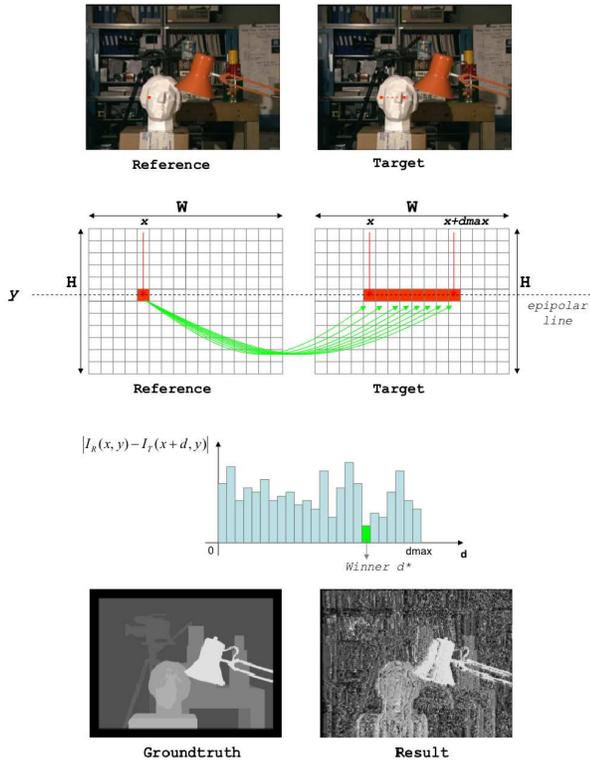


Figure 2: Illustration of the naive approach: The method takes a pixel from one image and compares it to possible matches in the other image; the pixel with the closest color similarity is declared a match.

In order to overcome some of the problems of the naive approach, cost-aggregation methods were proposed [12]. The

simplest of these methods are window-based, which apply the above mentioned naive approach with respect to a neighborhood of pixels (instead of a pixel-wise comparison). When considering an assignment to a pixel in the reference image, a window around this pixel is compared to other windows around relevant pixels in the target image. The displacement yielding the minimal window difference (using sum of squared differences, for example) is assigned as the pixel's disparity. The advantage of this approach over the naive method lies in the greater amount of data considered when assigning a certain disparity to a given pixel.

Window-based methods advocate a tradeoff between accuracy and the sharpness of the resulting disparity map. The larger the neighborhood is, the more information is used to accurately determine the disparity. However, this leads to blurry edges (see Figure 3), due to the attempt to incorporate information from both sides of the edge in determining the pixel's disparity. On the other hand, the smaller the neighborhood is, the less information is used and the less accurate the result is, although the sharper the edges are. (A very small neighborhood amounts practically to the naive approach covered above.)

It is possible to consider a three-dimensional matrix whose dimensions are the length and width of the images and the disparity range. Each matrix cell will contain the local cost associated with the disparity assigned to it. Simple cost-aggregation methods apply a blurring mechanism to incorporate data from neighboring pixels, under the assumption that the disparity map is smooth. Advanced cost-aggregation methods apply more sophisticated aggregation / blurring, such as the use of a bilateral filter which assigns more weight to neighboring pixels with the same color. This improves the performance since neighboring pixels with roughly the same color tend to belong to the same object, whose depth map is relatively smooth.

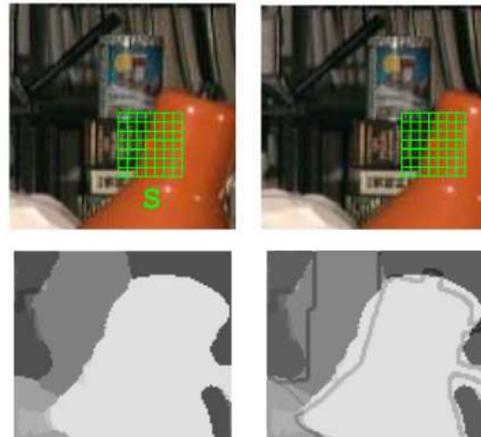


Figure 3: Illustration of tradeoff between disparity and edge sharpness in window-based approach; the larger the window size is, the more information is incorporated and the more blurry the edges are in the output stereo map.

Another cost-aggregation method is based on image segmentation [7]. This image processing operation splits an image into (fairly) homogeneous regions, each belonging to a single object whose depth map is assumed relatively smooth. Segmentation could have assisted in stereo matching, pro-

vided it was sufficiently accurate. Assuming exact object segments are given, cost-aggregation could be used with respect to each segment, independently, to compute more accurately the disparity value of a pixel. (Each segment is used as a kernel instead of a standard window.) Unfortunately, segmentation is still considered an open problem, and it usually results in “under-segmentation” / “over-segmentation” of objects and in “leaking” segments (see Figure 4). Also, the segmentation process is very sensitive to the parameters used and, in general, it is not robust to the input scene.

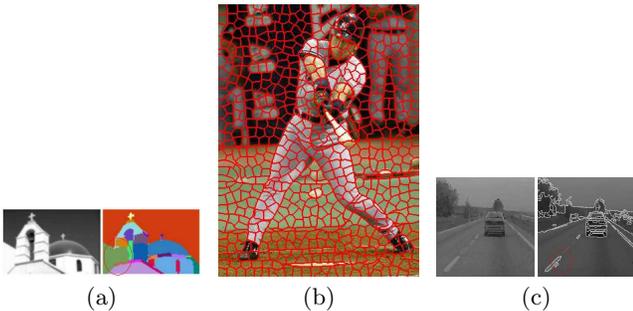


Figure 4: Illustration of segmentation artifacts [11]: (a) Under-segmentation, (b) over-segmentation, and (c) leaking segmentation.

The above discussed techniques determine a pixel’s disparity based on its local neighborhood, independently of the disparity of other pixels. To achieve better results, most state-of-the-art algorithms incorporate a global optimization scheme as a vital part of the algorithm [10, 13, 18]. The global optimization attempts to minimize an objective function that receives as input the disparity map and outputs a fitness score. A most notable objective function is of the form [3]

$$E_{Data}(DisparityImage) + E_{Smooth}(DisparityImage),$$

where  $E_{Data}(DisparityImage)$  denotes the local pixel disparity assignment cost (similarly to previously described local approaches) and  $E_{Smooth}(DisparityImage)$  is associated with the cost of maintaining a piecewise smooth disparity image (by adding a penalty inversely proportional to the smoothness of the disparity image).

The above formulation is common in various machine learning techniques, where we seek a fitting component (i.e.,  $E_{Data}$ ) and a regularization term (i.e.,  $E_{Smooth}$ ) to ensure an overall simple, non-overfitting solution. As in commonly used machine learning techniques, even the global minimum solution might be mistaken in comparison to the ground truth. The proximity of this model to ground truth is dependent on the objective function picked, in particular the regularization and data term.

As demonstrated in [3], minimizing this function can be mapped to the multi-way cut problem, which is known to be NP-complete [4]. Therefore, solving the problem is intractable (unless  $P = NP$ ). Several approximation techniques for this minimization problem were suggested [3, 6, 18]. Some of these approximation methods result in an objective function score that is not greater than a certain factor times the global minimum. However, this does not ensure an approximation bound regarding the disparity errors, since

satisfying the objective function used may not necessarily be compatible to ground truth.

A common objective function, according to the above formulation, sets the local data term to the absolute color difference between pixels in the reference image and their (hypothesized) counterparts in the target image. The smoothness term adds a constant for every spatially neighboring pixels having different disparities.

Another family of approximation algorithms uses dynamic programming to solve the minimization problem presented above [5, 16, 8]. Since solving the exact problem is intractable, these algorithms try to solve accurately a relaxed version of the problem, which is no longer NP-complete. Instead of solving the problem in a 2-D grid, the idea is to solve, independently, several relaxed sub-problems, each of which is concerned with the optimization of an objective function with respect to only part of the data, e.g., a 1-D row. Ignoring inter-line dependencies, one may thus apply repeatedly the common stereo scan-line technique [14] (which our method will attempt to improve) for solving the various (relaxed) 1-D problem instances. Optimization in the 1-D case is obtained due to Viterbi’s well-known algorithm [17].

The latter is a dynamic programming algorithm for finding the most likely sequence of hidden states. In this case, the hidden states are the disparity assignments for each pixel. The scan-line technique can be employed in the framework of Viterbi’s algorithm, with the possible disparities as the possible hidden states, a cost function for assigning a hidden state to a specific pixel, and a smoothness function penalizing two neighboring pixels on the line, provided their hidden states are unrelated.

The Viterbi algorithm maps the problem to finding the shortest path in a weighted graph generated as follows. Each pixel is represented by  $k$  vertices symbolizing its possible hidden states, i.e., the number of possible disparity assignments. Also, there is an edge between two vertices belonging to neighboring pixels, whose weight is given by the expression

$$E_{Data}(NextPixelDisparity) + E_{Smooth}(PrevPixelDisparity, NextPixelDisparity)$$

(Note that summing the above expression over all line pixels gives the objective function minimized by the scan-line method.) The graph consists also of two dummy vertices, the “Start” and “End” vertices. These vertices are connected to each of the labeled vertices corresponding to the first and last pixels in the chain, respectively, as shown in Figure 5. The data term and smooth term involving the dummy vertices are set to zero. It can be easily verified that each path from the Start vertex to the End vertex is associated with a sequence of hidden state assignments (i.e., disparity values), such that its corresponding total weight is the cost of the objective function (presented in [3]) for those specific hidden states (or disparities). Thus, the shortest path (from Start to End) yields the optimal disparity assignment for this relaxed problem instance.

Realizing that there are  $k^2$  edges between neighboring pixels, it can be shown that the time complexity of the scan-line algorithm is  $O(n \cdot k^2)$ , where  $n$  is the number of pixels in the chain and  $k$  is the number of possible disparity assign-

ments for each pixel. For some objective functions, the time complexity can be reduced to  $O(n \cdot k)$ .

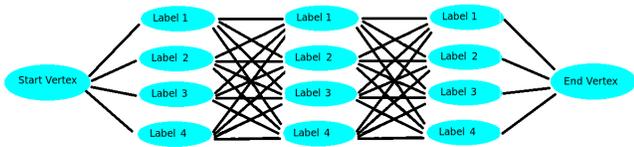


Figure 5: Graph for 1-D scan-line.

The 1-D scan-line technique ignores various constraints in order to achieve an efficient polynomial time solution. Considering each scan-line independently results, for example, in “striding” effects caused by the fact that no information from one scan-line is used to optimize another scan-line (see Figure 6). An important observation is that the less relaxed the problem is, the more suitable the solution becomes. In [16], a less relaxed version was treated by considering a tree induced by the image grid graph (i.e., a graph whose vertices are associated with the pixels and where an edge connects two vertices corresponding to 4-neighboring pixels), and optimizing the objective function with respect to this tree. The time complexity in this case was shown to be the same, i.e.,  $O(n \cdot k^2)$ .



Figure 6: Result due to vertical scan-line.

### 3. PROPOSED APPROACH

Our proposed approach explores the possibility of drawing on some of the above methods, in a relatively novel manner, for the stereo disparity problem. Specifically, we propose a hybrid GA-based optimization scheme, which combines the global search properties of genetic algorithms with the powerful properties of a local search.

Various components of a GA framework (such as initialization, crossover, mutation, and fitness function) will be specified by adapting relevant, previously described techniques. This adaptation focuses on the crossover operation, and involves effective “local” optimization that can cope with the very large dimensionality of the problem. (See detailed discussion below.) We believe that this hybrid approach will yield a practically sound scheme for the stereo matching problem.

## 3.1 Genetic Operators

This subsection describes the adaptation of each operator of the genetic scheme to fit the stereo correspondence problem, by incorporating also knowledge from the problem domain itself.

### 3.1.1 Chromosome Representation

Chromosome representation and encoding have a great impact on the performance of a genetic algorithm. A compact representation can significantly reduce the dimensionality and the search space. However, it also introduces difficulties in other aspects of the genetic algorithm, such as the crossover and fitness. A sparse image representation, on the other hand, such as contours or edge lists, requires sophisticated, time-consuming decoding and crossover mechanisms which often degrade the performance of the algorithm either by ignoring relevant data or by increasing significantly the algorithm’s time complexity.

A chromosome representation of the problem at hand (with respect to some set of pixels) consists merely of plain disparity values assigned to these pixels. The dimensionality is quite large and the search space contains  $DisparityRange^{|\text{Pixels}|}$  assignment possibilities. Despite this large search space, we show how the crossover routine can exploit domain knowledge in a clever manner to construct from this representation an improved solution.

### 3.1.2 Population Initialization

Selecting the initial population is important to the convergence of the genetic algorithm. In particular, a large diverse population of initial solutions would result in a broader perspective of the search space and would thus prevent convergence to a local minimum.

Our approach offers a compromise between a large diverse space and a small, highly adapted population. Natural images are mostly smooth with respect to the depth map, i.e., it is not likely for a pixel to have a different depth than its neighbor unless an edge is crossed. Therefore it is reasonable to assume that a bunch of spatially close pixels have roughly the same disparity value. We exploit this property to produce initial solutions by assigning a disparity value at random to a bunch of close pixels and repeating this process until no pixels are left unassigned. This creates a well-distributed initial population, and at the same time it enforces the assumption of a depth-smooth map by assigning the same disparity to small spatially contiguous regions.

### 3.1.3 Solution Mutation

This operator mutates the chromosomes, so as to generate new solutions that otherwise might not have been produced from the current population. Introducing mutation helps to diversify the population in each generation and avoid hopefully a local minimum. This operation is also guided by a tradeoff between less likely local minima and fast convergence, depending on the degree of mutation.

The “headless chicken” [9] was applied in our case. The crossover operation is applied to a given disparity map and a disparity map generated by the initialization described above. We consider the result from this crossover as a mutated chromosome. Using the hybrid crossover procedure as described in the next subsection ensures that the solution produced adds to the population diversity with little negative effect on the chromosome’s fitness. Consequently, the

time of convergence does not increase much in comparison to other mutation schemes.

This mutation procedure is applied to all chromosomes except the elite ones, that are passed as is to the next generation. Applying this mutation constantly to almost every chromosome (with mutation rate of 1.0), helps keep the population diverse and gain a better solution at the expense of losing valuable information. The likelihood of occurrences of the latter is smaller since the crossover operator incorporates knowledge of the domain.

### 3.1.4 Crossover Operator

The crossover operator is meant to merge two different solutions into a new solution in the next generation. The idea is that merging two solutions often evolves an improved solution. However, because of the curse of dimensionality in stereo matching, this is almost never the case. Thus, a powerful crossover technique is required.

The crossover proposed is based on the dynamic programming technique of the scan-line method. But instead of feeding the algorithm with all possible disparities, the disparity assignment to each pixel is limited to two possible values at most, one from the first solution and the other from the second. This procedure forces the algorithm to choose the globally optimal line assignments for each line, independently of another line, where an assignment is made either from the first or the second solution. Obviously, these assignments are optimal for each line independently, and they form a local/semi-optimal crossover solution.

Note that we choose horizontal and vertical scan-lines interchangeably (where the decision is made at random). Every two chromosomes are merged in a manner that is “optimal” with respect to the line direction chosen. Adding more lines in various orientations should improve the bottom-line result, as the crossover would accommodate many more orientations, i.e., it is more likely that the optimum of the global fitness (defined below) could be satisfied.

Finally, it is important to note that although the crossover is applied to global optimization, it is very efficient. Recall that the time complexity of the original scan-line algorithm is  $O(n \cdot k^2)$ . Since by construction, our crossover operator considers only two possible values (i.e.,  $k = 2$  in our case), its time complexity is linear in the number of pixels.

### 3.1.5 Solution Fitness

The fitness function is at the heart of a genetic algorithm’s scheme. It is used to measure the quality of a solution. The fitness has of course a great impact on the algorithm’s convergence and the solution accuracy.

We employ the well-known fitness function of the form

$$E_{Data}(DisparityImage) + E_{Smooth}(DisparityImage)$$

used by the scan-line algorithm. (The function was suggested in [3] and explained in Section 2.) The data term can simply be the absolute color difference between a pixel from the left image and its hypothesized corresponding pixel in the right image. The smoothness term between two neighboring pixels can be simplified to zero if they have the same disparity. Otherwise, if the absolute difference between the two disparities is 1, the smoothness term is assigned some  $\alpha > 0$ . If the disparity difference is greater than 1, the penalty is  $\beta$ , where  $\beta > \alpha$ .

Finding a global minimum to the fitness function yields a stereo disparity map whose agreement with the ground truth is highly dependent on the specific smoothness and data functions chosen. Even with the same functions above, the accuracy of the results could vary greatly for different values of  $\alpha$  and  $\beta$ . Finding a set of parameters that satisfy all images is difficult if not impossible. We managed to fix all of the parameter values for the benchmark datasets we experimented with (see Section 4). Although we obtained promising overall performance, the performance for each image pair could have been improved by fine tuning the parameter values to fit each of the datasets experimented with.

## 3.2 Intuitive Rationale

The dynamic programming scan-line algorithm is a commonly used stereo matching algorithm. It instantly provides the global minimum over a chosen line. Note, though, that the global minimum for each line is essentially a local minimum for the complete, non-relaxed fitness function, since inter-line constraints are not taken into account.

The proposed crossover procedure is applied to two possible solutions (i.e., chromosomes); it obtains a better local solution over a random line direction by applying a scan-line optimization in the direction picked, selecting – for each pixel – one of the (two) assignments offered by the parents. Although this building block may yield an improved solution over a specific line direction, it is restricted to local optima (global optima regarding the line direction picked), as it only considers the parents’ assignments and not the entire search space. However, by repeating the crossover procedure for various line orientations, it is expected that chromosomes will slowly improve with each orientation, and that a solution will eventually emerge that conforms to the relaxed fitness function over many line directions (meaning that if the solution “fits” many relaxed fitness functions with respect to many variations of partial data – this implies, hopefully, that it minimizes the non-relaxed fitness function over the entire data is minimized as well).

The crossover procedure quickly eliminates irrelevant assignment candidates, but it may eliminate relevant assignments as well, since they are not optimal over the line direction picked. To ensure that the global fitting assignments are not permanently removed, mutation is applied so that all of the assignments are considered. Furthermore, the fitness function used to score each chromosome is the complete non-relaxed function, which adds another “fail-safe” against the removal of pixel assignments that do fit the global function.

This procedure can be viewed as analogous to the well-known line search optimization [2]. In each phase of the algorithm, we optimize with respect to a specific line direction. The method’s progress is limited, since only two selected chromosomes are considered (i.e., not all possibilities). Integrating the line search approach with the GA framework tends to avoid local minima and achieve improved results.

The algorithm quickly converges to a solution despite the vast search space due to the local optimization used in our hybrid approach. The solution also yields higher accuracy over the generations due to the genetic algorithm’s mechanism of escaping local minima.

Table 1: Parameter values used by the hybrid algorithm

Algorithm Parameters	
Population size	80
Mutation probability	1.0
Elitism	3
Epochs	500
$\alpha$	4
$\beta$	10

#### 4. EMPIRICAL ANALYSIS

The performance of our approach was assessed on the Middlebury benchmark site [1], which is well recognized and widely used by the computer vision community. The site contains standard datasets for researchers to experiment with using an on-line interface. Also, it contains results for the most current algorithms and their relative ranking.

Figures 7—10 below present some preliminary results. The top three images in each figure are the left input image, right input image and ground truth (of the disparity map). Also, the bottom two images in each figure (from left to right) correspond to the disparity maps obtained (on the Middlebury site) by our approach and the dynamic programming scan-line approach (we have set out to improve). All of the performance evaluations were conducted on-line on the Middlebury evaluation site [1], according to the benchmark protocol and the test datasets (of 4 stereo pairs) agreed upon. The error rates obtained in this evaluation were 13.5% and 16.6%, respectively, for our approach and the scan-line method. Thus, the number of pixels assigned with a disparity different from the ground truth decreased by about 20% in comparison with the original scan-line method. Note that all disparity maps were obtained for the same parameter values of Table 1, used in all of the evaluation test data on the Middlebury site.

A main evaluation criterion used is the number of incorrect pixels, i.e., pixels for which the absolute disparity difference between the solutions provided and the ground truth is greater than 1. Table 2 shows the percentage of incorrect pixels obtained via the Middlebury evaluation site for our approach and the scan-line optimization method over non-occluded areas.

Table 2: Comparison of incorrect pixels, according to the Middlebury evaluation site for our approach vs. the scan-line algorithm

	Tsukuba	Venus	Teddy	Cones
Our approach	2.98%	2.71%	13.9%	6.53%
Scanline Opt	5.08%	9.44%	19.9%	13.0%

#### 5. CONCLUSION

We presented in this paper a novel scheme for the stereo matching problem. The algorithm presented yields a significant improvement in accuracy relative to the original scan-line approach it is based upon. In particular, the empirical results show a relative improvement of roughly 20%. Our scheme also enables to “boost” a weak local optimization scheme via a genetic algorithm. Specifically, it could replace scan-line optimization modules in state-of-the-art algo-

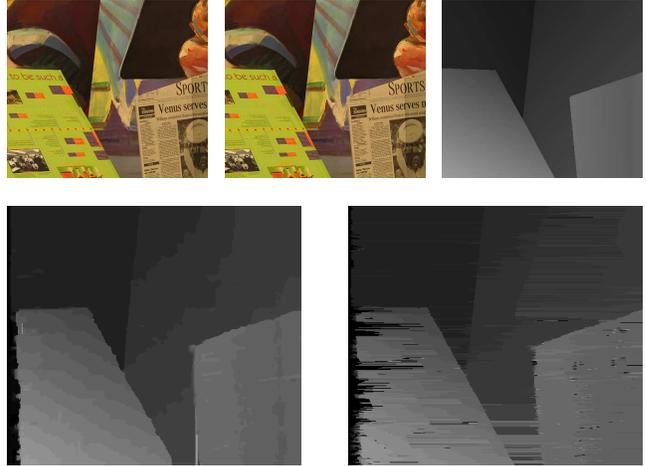


Figure 7: Top row shows the Venus stereo pair and its ground truth; bottom row shows (from left to right) the resulting disparity maps of our approach and the original scan-line algorithm)

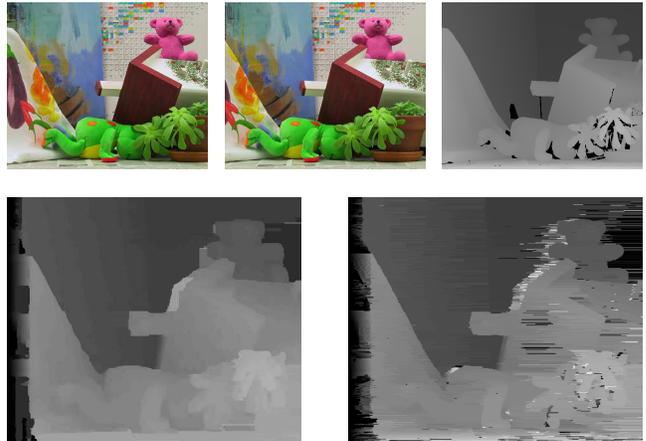


Figure 8: Top row shows the Teddy stereo pair and its ground truth; bottom row shows (from left to right) the resulting disparity maps of our approach and the original scan-line algorithm

rithms (that achieve higher accuracy than that of our hybrid algorithm) [15, 13] to further enhance their performance.

Further research regarding the fitness function can contribute a great deal to the accuracy of the algorithm. Also, it would be of interest to investigate the impact on the convergence rate of the algorithm and the overall performance when taking in consideration the k-shortest path in the crossover operator.

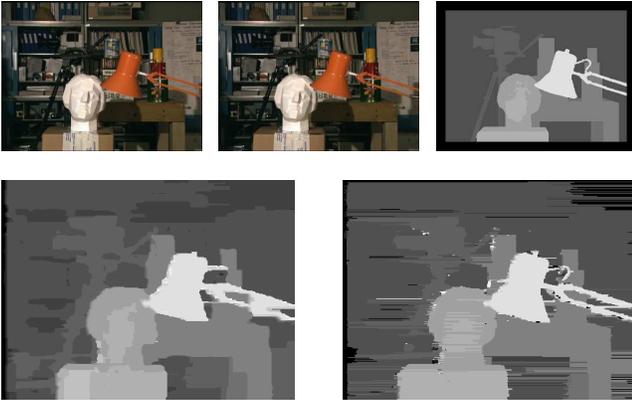


Figure 9: Top row shows the Tsukuba stereo pair and its ground truth; bottom row shows (from left to right) the resulting disparity maps of our approach and the original scan-line algorithm

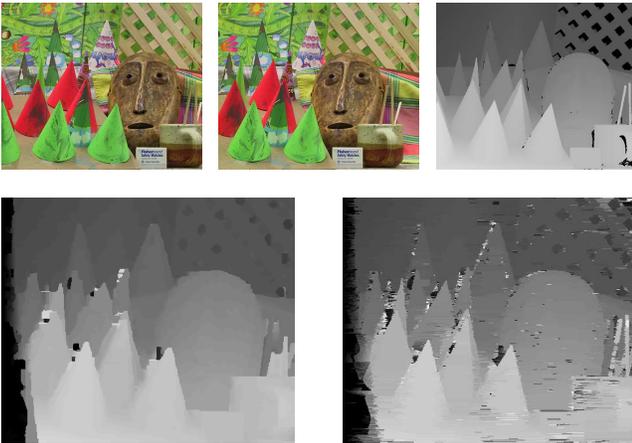


Figure 10: Top row shows the Cones stereo pair and its ground truth; bottom row shows (from left to right) the resulting disparity maps of our approach and the original scan-line algorithm

## 6. REFERENCES

- [1] Middlebury stereo vision page (<http://vision.middlebury.edu/stereo>).
- [2] M.J. Box, D. Davies, and W.H. Swann. Non-linear optimization techniques. *Mathematical and statistical techniques for industry monograph*, 5:60, 1969.
- [3] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *In Proceedings of the IEEE International Conference on Computer Vision*, pages 377–384, 1999.
- [4] E. Dahlhaus, D.S. Johnson, C.H. Papadimitriou, P.D. Seymour, and M. Yannakakis. The complexity of multiway cuts. *In Proceedings of the ACM Symposium on Theory of Computing*, pages 241–251, 1992.
- [5] Y. Deng and X. Lin. A fast line segment based dense stereo algorithm using tree dynamic programming. *In Proceedings of the European Conference on Computer Vision*, 3953:201–212, 2006.
- [6] P.F. Felzenszwalb and D.P. Huttenlocher. Efficient belief propagation for early vision. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 70:41–54, 2004.
- [7] Mark Gerrits and Philippe Bekaert. Local stereo matching with segmentation-based outlier rejection. page 66, 2006.
- [8] H. Hirschmuller. Stereo vision in structured environments by consistent semi-global matching. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2:2386–2393, 2006.
- [9] T. Jones. Crossover, macromutation, and population-based search. *In Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 73–80, 1995.
- [10] A. Klaus, M. Sormann, and K. Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. *In Proceedings of the International Conference on Pattern Recognition*, 3:15–18, 2006.
- [11] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. *In Proceedings of the IEEE International Conference on Computer Vision*, 2:416–423, 2001.
- [12] S. Mattoccia, E. Addimanda, and L. Di Stefano. Classification and evaluation of cost aggregation methods for stereo correspondence. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.
- [13] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang. On building an accurate stereo matching system on graphics hardware. *In Proceedings of the International Conference on Computer Vision Workshops*, pages 467–474, 2011.
- [14] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47:7–42, 2002.
- [15] Xun Sun, Xing Mei, Shaohui Jiao, Mingcai Zhou, and Haitao Wang. Stereo matching with reliable disparity propagation. *In proceedings of the IEEE International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, pages 132–139, 2011.
- [16] O. Veksler. Stereo correspondence by dynamic programming on a tree. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2:384–390, 2005.
- [17] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13:260–269, 1967.
- [18] Z. Wang and Z. Zheng. A region based stereo matching algorithm using cooperative optimization. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.
- [19] W.Y. Yau and H. Wang. Fast relative depth computation for an active stereo vision system. *Real-Time Imaging*, 5:189–202, 1999.