International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622
International Conference On Quality Up-gradation in Engineering, Science & Technology
(IC-QUEST- 11th April 2015)

RESEARCH ARTICLE OPEN ACCESS

Stereo Matching of High Resolution Images

Apurva Naik, Gourang Mulay, Arti Khaparde, Pranoti Dhole
Department of Electronics and Telecommunication
Maharashtra Institute of Technology
Pune, Maharashtra, India
Emails :{apurva.naik,gourang.mulay,arti.khaparde} @mitpune.edu.in; pranotidhole4@gmail.com

Abstract-
Stereo vision is one of the most heavily researched topics in computer vision, and much of the progress over the last decade has been driven by the availability of standard test images and benchmarks.In thispaper we present a system for generating a disparity map for new high-resolution two view datasets using structured lighting. These two views of the data set are segmented using Particle Swarm Optimization(PSO), Darwinian Particle Swarm Optimization (DPSO) and fractional order Darwinian Particle Swarm optimization (FO-DPSO) algorithms. Segmented Stereo images are used to generate disparity map which gives depth levels in image at par with the original image. Subjective quality of 3-D image generated after application of these algorithms is better for FO-DPSO algorithm as compared to PSO and DPSO algorithms.

Keywords-Particle Swarm Optimization(PSO); Darwinian Particle Swarm Optimization(DPSO); Fractional Order Darwinian Particle Swarm Optimization(FO-DPSO).

I. Introduction

A two-dimensional camera image does not give information about depth levels. However, information about depth is required in several applicationssuch as, satellite imaging, robotic vision, target tracking and automatic map making. Objective of proposed system is to reduce time for stereo matching for high resolution images from Middlebury [2] dataset. Stereo matching is used to extract depth informationfrom images [3].Until recently; stereography was used either for entertainment purpose or DEM (Digital Elevation Model).

Proposed system for disparity estimation and 3-D generation is shown in fig.1.Data from Middlebury data set is input to the biologically inspired segmentation algorithms
PSO, DPSO, FO-DPSO .Resulting input image pairs are given to the stereo matching algorithm. Output of stereo matching algorithm will be compressed 3-D image and disparity map which can give estimation of depth levels in image.

The rest of this paper isorganized as follows. Section II describes the biologically inspired segmentation algorithms. Section III describes the stereo matching algorithms. Section IV gives details of results. Section V concludes the paper.

II. Biologically Inspired Segmentation Algorithms

The human identifies the objects by analyzing features of the objects such as color, texture and shape. Similar techniques can be applied for computer vision.Segment-based methods have attracted attention due to their good performance on handling boundaries and texture less regions.

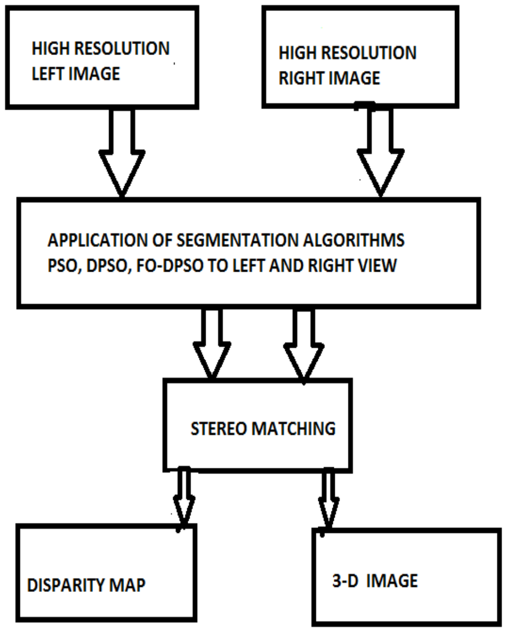

HIGH RESOLUTION LEFT IMAGE
HIGH RESOLUTION RIGHT IMAGE
APPLICATION OF SEGMENTATION ALGORITHMS PSO, DPSO, FO-DPSO TO LEFT AND RIGHT VIEW
STEREO MATCHING
DISPARITY MAP
3-D IMAGE

Figure 1. Block Diagram of proposed system

Segment based methods are based on the assumption that the scene structure can be approximated by a set of non-overlapping planes in the disparity space and that each plane of target image is coincident with at least one homogeneous color segment in the reference image. Segment based methods perform well in reducing the ambiguity associated with texture less regions and enhancing noise tolerance.
The computational complexity is reduced due to much larger segments. Matching becomes much easier even in the presence of noise, intensity variation and slight deviations in segmented area. Noise tolerance is improved by aggregating over pixels with similar colors. Small segments may be insufficient for estimating surfaces like slanted planes, while large segments may containsegmentation errors which can affect the accuracy of disparity estimation. Similar colors in image do not always mean similar disparity. For example, the projected image area of a very slanted Lambertian surface with uniform texture tends to be classified as one image segment, while the depths may change a lot within this segment. Numeroussegmentation techniques are present to divide the image into segments and then matching is carried out on these segments.
Segment-based stereo matching generally performs four consecutive steps. First, segment the reference images using robust segmentation method; second, get initial disparity map using local match method; third, a plane fitting technique is employed to obtain disparity planes. Finally, optimal disparity plane assignment is approximated by using optimization methods [4].
Following segmentation algorithms are applied prior to matching i) PSO ii) DPSOiii) FO-DPSO.
For each of these algorithms, we examine three characteristics:

1. ***Correctness***: the ability to produce segmentations which agree with human intuition.
2. ***Stability with respect to parameter choice***: the ability to produce segmentations of consistent correctness for a range of parameter choices.
3. ***Stability with respect to image choice***: the ability to produce segmentations ofconsistent correctness using the same parameter choice on a wide range of different images.

All these algorithms come under clustering based segmentation technique. Till date, lots of image segmentation algorithms exist and be extensively applied in science and daily life. A definition of clustering is process of organizing objects into groups whose members are similar in some way.
Recently, researchers are interested in locating multiple local optima of a given multi-modal function in a d –dimensional search space. For this purpose nature inspired techniques are used.
Several biologically or nature inspired algorithms have been explored for image segmentation [5-6].The computational time and fitness value are most important indicators for the algorithms. Moreover, in high resolution image applications, using high speed algorithm is main objective [7]. The state of the art is to employ Swarm-Based collective intelligence to image.

1) Image Segmentation By Particle Swarm Optimization

Particle swarm optimization (PSO) is an optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995, inspired by social behavior of bird flocking or fish schooling. The particle swarm concept originated as a simulation of simplified social system. The original intent was to graphically simulate the choreography of bird of a bird flock or fish school. However, it was found that particle swarm model can be used as an optimizer. Consider the following scenario: a group of birds are randomly searching food in an area. There is only one piece of food in the area being searched. All the birds do not know where the food is. But they know how far the food is, in each iteration. Sowhat's the best strategy to find the food? The effective one is to follow the bird which is nearest to the food. PSO learned from the scenario and used it to solve the optimization problems. In PSO, each single solution is a "bird" in the search space. We call it "particle". All of particles have fitness values which are evaluated by the fitness function to be optimized, and have velocities which direct the flying of the particles. The particles fly throughthe problem space by following the current optimum particles [8].Flowchart of basic PSO algorithm is as shown in Fig.2.
PSO is initialized with a group of random particles (solutions) and then searches for optima by updating generations. For every iteration, each particle is updated by following two "best" values. The first one is the best solution (fitness) it has achieved so far. This value is called pbest.Another "best" value that is tracked by the particleswarm optimizer is the best value, obtained so far by any particle in the population.

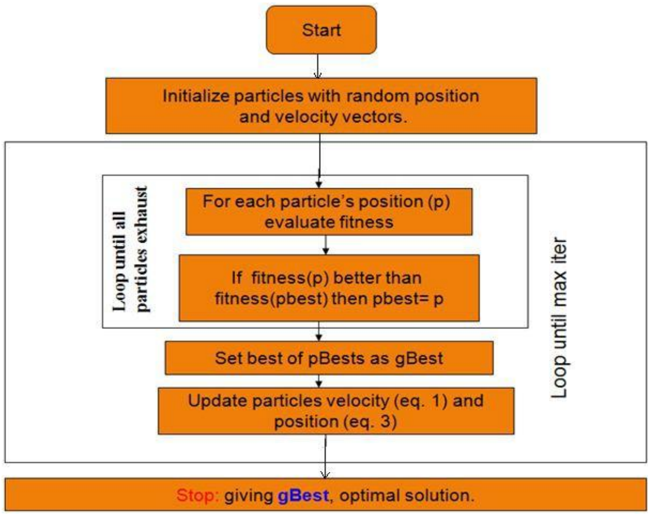

Start
Initialize particles with random position and velocity vectors.
Loop until all particles exhaust
For each particle's position (p) evaluate fitness
If fitness(p) better than fitness(pbest) then pbest= p
Loop until max iter
Set best of pBests as gBest
Update particles velocity (eq. 1) and position (eq. 3)
Stop: giving gBest, optimal solution.

Figure 2. Flowchart for Basic PSO Algorithm

This best value is a global best and called gbest. When a particle takes part of the population as its topological neighbors, the best value is a local best and is called lbest.

After finding the two best values, the particle updates its velocity and positions with following equations (1) and (2)

$$V_R = W * V_R + rand_1 * (r_1 * (X_{best} - X_R)) + rand_2 (r_2 * (g_{auxR} * g_{best} - X_R) \qquad (1)$$

$$X_R = X_R + V_R (2)$$

$V_R$is the particle velocity; ***X_R***is the current particle (solution). **pbest**and **gbest**are defined as stated before. rand( )is a random number between 0 and 1. r_1, r_2 are learning factors. Usually ***$r_1 = r_2 = 2$.*** 'gbest' is obtained as follows.

$$g_{best} = \frac{X_R \sum probR\,[1{:}X_R(j,1)] * \sum_1^{X_R(j,1)} X_{prob}\; R[1{:}X_R(j,1)]}{X_R \sum probR\,[1{:}X_R(j,1)] - \sum_1^{Lmax} (probR\,(1{:}Lmax))^2} \qquad (3)$$

For a known (differentiable) function f, calculus can easily provide us with the minima and maxima of f. However, in real-life optimization tasks, this objective function f is often notdirectly known. Instead, the objective function is a "black box" to which we apply parameters (the candidate solution) and receive an output value. The result of this evaluation of a candidate solution becomes the solution's fitness. The final goal of an optimization task is to find the parameters in the search space that maximize or minimize this fitness.

The PSO algorithm works by simultaneously maintaining several candidate solutions in the search space. During each iteration of the algorithm, each candidate solution is evaluated by the objective function being optimized, determining the fitness of that solution. Each candidate solution can be thought of as a particle "flying" through the fitness landscape finding the maximum or minimum of the objective function. Initially, the PSO algorithm chooses candidate solutions randomly within the search space. The PSO algorithm simply uses the objective function to evaluate its candidate solutions, and operates upon the resultant fitness values.

Each particle maintains its position, composed of the candidate solution and its evaluated fitness, and its velocity. Additionally, it remembers the best fitness value it has achieved so far during the operation of the algorithm, referred to as the individual best fitness, and the candidate solution that achieved this fitness, referred to as the individual best position or individual best candidate solution. Finally, the PSO algorithm maintains the best fitness value achieved among all particles in the swarm, called the global best fitness, and the candidate solution that achieved this fitness, called the global best position or global best candidate solution. While maximum iterations or minimum error criteria is not attained, Particles' velocities on each dimension are clamped to a maximum velocity V_{max}. If the sum of accelerations would cause the velocity on that dimension to exceed V_{max}, which is a parameter specified by the user, then the velocity on that dimension is limited to V_{max}.

There are two key steps when applying PSO to optimization problems: the representation of the solution and the fitness function. One of the advantages of PSO is that PSO take real numbers as particles. It is not like GA, which needs to change to binary encoding, or special genetic operators have to be used. For example, to find the solution for $f(x) = x1^2 + x2^2 + x3^2$, the particle can be set as (x1, x2, x3), and fitness function is f(x). Then the standard procedure can be used to find the optimum. The searching is a repetitive process, and the stop criterion is that either maximum iteration number is reached or theminimum error condition is satisfied.

A list of parameter values to be selected is given below.

- The number of particles: the typical range is 20 - 40. Actually for most of the problems 10 particles is large enough to get good results. For some difficult or special problems, one can try 100 or 200 particles as well.
- Dimension of particles: It is determined by the problem to be optimized,
- Range of particles: It is also determined by the problem to be optimized, you can specify different ranges for different dimension of particles.
- V_{max}: it determines the maximum change one particle can take during each iteration. Usually, we set the range of the particle as

the V_{max} for example, the particle (x1, x2, x3) X1 belongs [-10, 10], then $V_{max} = 20$

Learning factors: r_1 and r_2 usually equal to 2 to assign equal weight to both terms in velocity equation. However, other settings were also used in different papers. But usually r_1 equals to r_2 and ranges from (0, 4).

2) Image Segmentation using Darwinian PSO

The computational time is one of the most important indicators along with fitness value which determine the ability of the algorithm. In real-time applications, using a high speed algorithm is the main objective.

A general problem with the PSO and other optimization algorithms is that of becoming trapped in a local optimum .In search of a better model of natural selection using the PSO algorithm, the DPSO was formulated [8] in which many swarms of test solutions may exist at any time. Each swarm individually performs just like an ordinary PSO algorithm with some rules governing the collection of swarms that are designed to simulate natural selection.

The traditional PSO-is compared with the DPSO-based segmentation method to determine the n-1 optimal n-level threshold on a given image. Each swarm individually performs just like an ordinary PSO algorithm in which natural selection (Darwinian principle of survival of the fittest) is used to enhance the ability to escape from local optima. When a search tends to a local optimum, the search in that area is simply discarded and another area is searched instead. In this approach, at each step, swarms that get better are rewarded (extend particle life or spawn a new descendent) and swarms which stagnate are punished (reduce swarm life or delete particles). To analyze the general state of each swarm, the fitness of all particles is evaluated and the neighborhood and individual best positions of each of the particles are updated. If a new global solution is found, a new particle is spawned. A particle is deleted if the swarm fails to find a fitter state in a defined number of steps

a) Natural Selection in DPSO

The basic assumptions made to implement Darwinian PSO are:

- The longer a swarm lives, the more chance it has of possessing offspring. This is achieved by giving each swarm a constant, small chance of spawning a new swarm.
- A swarm will have its life-time extended (be rewarded) by finding a more fit state.
- A swarm will have its life time reduced for failing to find more fit state.

In nature, individuals or groups that possess a favorable adaptation are more likely to thrive and procreate. The favorable adaptation is assumed to prolong the lifetime of the individual. Unfavorable adaptations shorten the lifespan of an individual or group. Particle birth and death within a swarm and swarm birth and death within the collection of swarms must be characterized.

b) Particle and swarm initialization:

Each PSO particle is an array of N numbers; the array could contain a binary string. The choice of the domain of the particle array elements, $x_i$as well as the encoding of the test solution as an array of numbers is motivated by the particular optimization problem. Each dimension of each particle is randomly initialized on an appropriate range $x_{min} \leq x_i \leq x_{max}$. The velocities are also randomly initialized on a range, $v_{min} \leq v_i \leq v_{max}$, that allows particles to traverse a significant fraction of the $x_i$in a single iteration when moving at $|v_i| = v_{max}$. Note that when a particle is created, its velocity is randomized to encourage exploration. Each swarm is initialized with a population of particles. At each step of the algorithm, labeled Main Program Loop in the pseudo code below, the Evolve Swarm algorithm, operates on each swarm. After evolving each swarm,each is allowed to spawn a new swarm with a fixed probability. After spawning, the selection process is executed. All swarms that are no longer progressing are deleted. To evolve an individual swarm, the fit nesses of all of the particles in the swarm are evaluated.

The neighborhood and individual best positions of each of the particles are updated. The swarm spawns a new particle if a new global best fitness is found. A particle is deleted if the swarm has failed to find a more fit state in an allotted number of steps. A swarm's particle population, m is bounded such that, $m_{min} \leq m_i \leq m_{max}$When aswarm's m_{min} population falls below , the swarm is deleted. DPSO algorithm is indicated in table 1.

Table 1 DPSO Algorithm

Main Program Loop	**Evolve Swarm Algorithm**
For each swarm in the collection	For each particle in the swarm **Update Particle Fit nesses'**
Evolve the swarm (Evolve Swarm Algorithm: right)	For each particle in the swarm **Update Particle Bests**
For each swarm in the collection	For each particle in the swarm **Move Particle**
Allow the swarm to spawn	If swarm gets better **Reward swarm :**
Delete "failed" swarms	spawn particle : extend swarm life
	If swarm has not improved **Punish swarm :** Possibly delete particle : **reduce swarm life**

The number of times a swarm is evolved without finding an improved fitness is tracked with a search counter SC. If the swarm's search counter exceeds a maximum critical threshold, SC_c^{max}, a particle is deleted from swarm. When a swarm is created, its search counter is set at zero. When a particle is deleted, the swarm's search counter is reset not to zero but to a value approaching SC_c^{max}as the time during which the swarm makes no improvement in fitness increases. The purpose of this reduction in tolerance for stagnation is to try to maintain a collection of swarms that are actively improving. If $N_{kill.}$ is the number of particles deleted from the swarm over a period in which there is no improvement in fitness, then the reset value of the search counter is chosen to be

$$SC_c(N_{kill}) = SC_c{}^{max}\left[1 - \frac{1}{N_{kill}+1}\right]$$

c) Condition for spawning particles and swarms

At each step, each swarm may spawn a new swarm. To be able to spawn a new swarm, an existing swarm must have N_{kill}. = 0. If this condition is met and the maximum number of swarms will not be exceeded, the swarm spawns a new swarm. with probability p=f/Ns, where f is a uniform random number on [0,1] and Ns is the number of swarms. The purpose of the factor of 1/Ns is to suppress swarm creation when there are large numbers of swarms in existence. When a swarm spawns a new swarm, the spawning swarm (parent) is unaffected. To form the spawned (child) swarm, half of the particles in the child are randomly selected from the parent swarm and the other half are randomly selected from a random member of the swarm collection (mate). The spawned or child swarm may inherit other attributes from either parent or mate as necessary to design experimentation for the Darwinian PSO algorithm. A particle is spawned whenever a swarm achieves a new global best fitness. The results obtained after DPSO segmentation of Tsukuba image are shown in figure 3.5. Table 2 shows parameter values selected for PSO, DPSO and FO-DPSO algorithm.

3)Image Segmentation using Fractional Order Darwinian PSO (FO-DPSO)

Some optimization methods are fast but not efficient (for finding the global optimum) and vice versa. It has been recently proved in [9] for benchmarking optimization problems that the FODPSO is faster than the PSO (the most well-known optimization algorithm in terms of speed)and more efficient than the DPSO (in order to find the global optimum while avoiding local optima). Therefore, applying the FODPSO to the segmentation of images may allow achieving bothvital important goals at once. More specifically, due to its convergence speed, thisoptimization method may present itself as a potential solution to a wide variety of complex problems .

Table 2: Parameters values selected for PSO, DPSO, FO-DPSO Algorithm

Selection of Initial Parameter Values			
Parameter	**PSO**	**DPSO**	**FO-DPSO**
Population(N)	150	30	30
Number of iteration(I_T)	150	150	150
Individual weight of particles $Ø_1$	0.8	0.8	0.8
Social Weight of particles $Ø_2$	0.8	0.8	0.8
Inertial Factor	1.2	1.2	1.2
V_{min}	-1.5	-5	-5
V_{max}	1.5	5	5
N_s	-	4	4
N_{min}(Minimum Population of swarm)	-	10	10
N_{max}(Maximum Population of Swarm)	-	50	50
N^s_{min}(Minimum Number of Swarms)	-	2	2
N^s_{max}(Minimum Number of Swarms)	-	6	6
N_{kill} (Total Number of Swarms Deleted)	-	10	10
α (A fractional Coefficient)	-	-	0.2

III. Stereo matching

Disparity refers to the difference in image location of an object seen by the left and right eyes, resulting from the eyes' horizontal separation (parallax). The brain uses disparity to extract depth information from retinal images in stereopsis. In computer vision disparity refers to the difference in coordinates of similar features within two stereo images .

Considering a single pixel in left image, and to compute its correspondence in the right image a variety of search techniques can be used to match pixels based on their local appearance as well as the motions of neighboring pixels. In the case of stereo matching some additional information is available, namely the positions and calibration data for the cameras that took the pictures of the same static scene. This information can be utilizedto reduce the number of potential correspondences, and hence speed up the matching and increase the reliability of matching

Initially we use stereo pair and images in standard form, i.e. with corresponding epipolar lines lying on corresponding image scan lines. Suitable transformation, known as rectification needs to be applied to obtain a pair ofimages in standard form from the original ones. Our algorithm is based on segmentation based stereo matching which givesdense disparity map. Novelty of our algorithm is stereo matching is carried out on compressedstereo images forgeneration of dense disparity. Most stereo matching algorithms today focus on dense correspondence; since this is required for many applications. This problem ismore challenging than sparse correspondence. Inferring depth values in texture less regions requires a certain amount of guesswork. Categorization scheme for dense correspondence algorithms was first proposed by [1]. The taxonomy consists of a set of algorithmic "building blocks" from which a large set of algorithms can be constructed. Stereo algorithms generally perform the following four steps:

- matching cost computation;
- cost (support) aggregation;
- disparity computation and optimization
- Disparity refinement.

A. Simple winner take all Algorithm

Disparity is computed as shown in fig. 3. As a result, we get three sets of disparity cost. Optimization of these 3 sets is done by using winner-take-all method. This method inspects the cost associated with each disparity set via window centered on pixel. Disparity with smallest aggregated cost is selected and given as estimated disparity map.

Disparity estimation was done for different sets of image pair as follows

1. Left and right original images
2. Left and right segmented images using mean shift algorithm, K-means, PSO, DPSO, FO-DPSO.

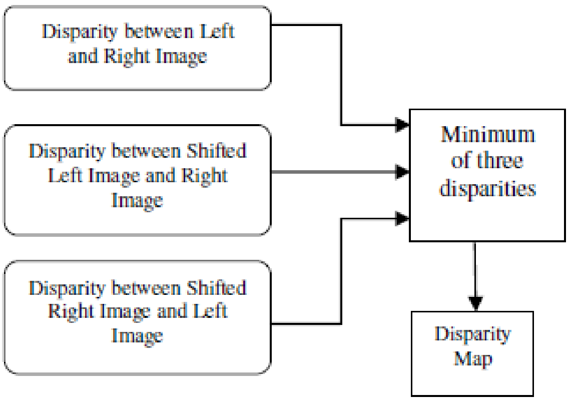

Disparity between Left and Right Image
Disparity between Shifted Left Image and Right Image
Disparity between Shifted Right Image and Left Image
Minimum of three disparities
Disparity Map

Figure 3.Process of disparity estimation.

Our cost function assumes brightness constancy for correspondingpixels. Our second method is global method which uses sum of absolute differences pixel wise.

B. Region based Disparity Estimation Algorithm by Disparity Growing

1) Minimizing the Error Energy

In this method, we used block-matching technique in order to construct an Error Energymatrix for everydisparity. Let's denote left image in RGB formatby $L(i, j, c)$,and denote right image in RGB format by $R(i, j, c)$ anderror energy by $e(i, j, d)$. For $n \times m$ window size of block matching, error energy $e(i, j, d)$ can be expressed by

$$e\ (i,j,d) = \frac{1}{3.n.m} + \sum_{x=i}^{i+n}\sum_{y=j}^{j+m}\sum_{k=1}^{3}\big(L(x,y+d,k) - Rx,y,k2 \quad (4)$$

Where, c represents RGB components of images and takes value of {1,2,3} corresponding to red, blue and green. 'd' is the disparity. For a predetermined disparity search range (w), every e (i, j, d) matrix respect to disparity is smoothed by applying averaging filter many times. Averaging filter (linear filter) removes very sharp change in energy which belongs to incorrect matching. Another important property of repeating application of averaging filter is that it makes apparent global trends in energy. Considering global trend in error energy naturally makes this algorithm a region based algorithm. For $n \times m$ window size, averaging filtering of $e(i, j, d)$ can be expressed by following equation

$$\text{e(i, j,d)} = \frac{1}{n.m}\sum_{x=i}^{i+n}\sum_{y=j}^{j+m} e(x,y,d) \qquad (5)$$

After iterative application of averaging filtering to error energy for each disparity, the disparity (d) is selected, which has minimum error energy ẽ(i, j, d) as the most reliable disparity estimation for pixel (i, j) of disparity map.

2)Line Growing

a)*Root Selection process:*

Select a point, which does not belong to any grown region and find its disparity using energy function equation (4.6). Set it as root point and set its disparity to region disparity then go to step 2. If point with lower enough error energy is not found repeat this step for the next point.

b)*Region Growing process:*

Calculate error energy of neighboring points just for root point disparity, which was called region disparity. If it is lower than the predetermined error energy threshold, associate this point to region.

Proceed for the Step 2 until region growing is possible. In the case that region growing is completed, go to step 1 to find out new root point to repeat these steps. When all points in image are processed, algorithm is stopped. Grown disparity regions compose the disparity map d (i, j).

c)*Depth Estimation*

Depth estimation is an important tool in several applications such as machine vision, robotics, and satellite terrain mapping. With recent advances in 3D consumer video communications technology, application of depth estimation is likely to grow significantly in near future. One of the objectives of this work is the depth estimation. The depth estimation techniques are of two types. Depth estimation using Laser or infrared ranging techniques are precise and popular. However, their applications are limited for certain tasks. For example, it is not advisable to laserscan a live human. Stereoscopic methods, on the other hand, are purely passive and uses a pair of cameras (left and right) to map a scene. The lateral shift between the images captured by the two cameras i.e. Disparity, is used to estimate the depth of different parts of a scène. Disparity estimation gives good results for finding short distances. Disparity of each pixel is calculated through the stereo matching algorithm. Using stereo camera parameters from calibration and the disparity between corresponding stereo points, depths in the stereo images can be retrieved. In order to find corresponding pairs of stereo points, they first have to be compared for different disparities, after which the best matching pairs can be determined. The maximum range at which the stereo vision can be used for detecting obstacles depends on the image and depth resolution. Absolute differences of pixel intensities are used in the algorithm compute stereo similarities between points.Using equation 6 below depth for each pixel position is calculated.

$$Z = b\frac{f}{d} \qquad (6)$$

There are only three parameters required to find depth or distance from disparity. The location of photo receptors of camera is called image plane. The focal length is distance between photo Receptor and lens which is specified in the camera data sheet as **f**. Base line width **b** is separation between stereo cameras and **d** is the disparity of each pixel. Measurement of X and Y locations in the real view are carried out with the help of yard stick or measuring tape and it is compared with the (x, y) pixel position on the camera. It is mapping of physical quantity in cm or meter to pixel scale. Due to this mapping all cm values are converted into pixel values and 3-D world co-ordinates of points corresponding to each pixel can be constructed from disparity map.

IV .Results

A.Results on the Middlebury Data Set 2014

1)Comparison on the basis of execution time

In version 3 of the Middlebury stereo evaluation important changes are made and added many new features, integrating lessons learned from previous evaluations. Data Sets have high resolution. Data sets are provided with three resolutions F-full, H-half, and Q-quarter. All the segmentation algorithms are applied to these high resolution images and very good results are obtained. Fig. 4 shows graph of time required for segmentation using three variants PSO, DPSO, FO-DPSO.

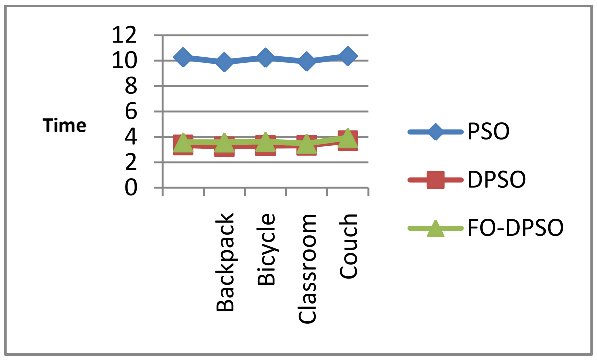


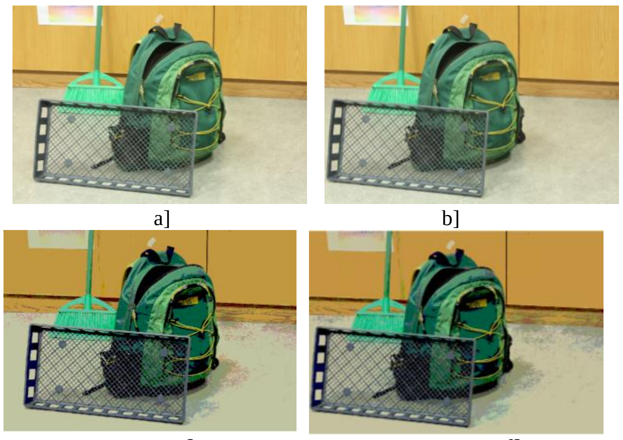
Figure 4. Graph of time required for segmentation using three techniques.

It can be observed that DPSO and FO-DPSO are efficient if time for segmentation is considered. Results after segmentation by above mentioned techniques for Backpack image is shown in fig.5.Fig. 5a] and 5b] represent original left view and right views. Fig .5c],and 5d] represent images obtained using PSO segmentation.Fig.5e] and 5f] represent images obtained through DPSO segmentation. Fig 5g] and 5h] represent images obtained using FO-DPSO segmentation.

Table 3 shows time required for stereo matching using Winner Take All algorithm

Table 3 Time required for stereo matching using Winner Take All

Image name	*Original Disparity*	*PSO*	*DPSO*	*FO-DPSO*
Adirondack	6.885001	4.297866	4.36681	4.288686
Backpack	6.319659	4.02989	4.18308	4.046667
Bicycle	8.59833	5.181148	5.13556	5.080545
Classroom	55.100283	4.536314	4.5496	4.524978
Couch	7.879685	3.08146	3.04838	3.080999



a] b]

c] d]

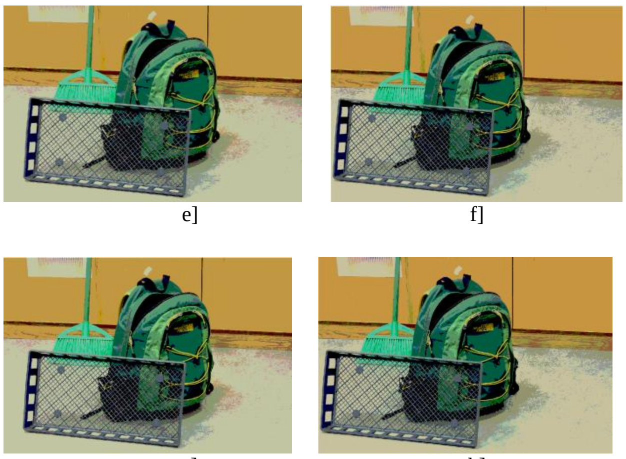


Figure 5 Images obtained through various segmentation Techniques

It can be observed from graph of fig. 6 that time required for segmented stereo matching is less compared to original images using all the three techniques. Y axis shows log scale.
Fig.6 shows graph of execution time in seconds for matching of high resolution images using Winner Take All algorithm. The peak in the graph is obtained due to high resolution classroom image.
Table 4 shows time required for stereo matching using line growing algorithm. Since line growing algorithm uses disparity refinement step there is significant increase in time required for stereo matching of even segmented images .Compared to original image matching time still segmented images requires less time compared using line growing technique also.Graph of execution time for various segmentation techniques using line growing algorithm is shown in fig. 7.

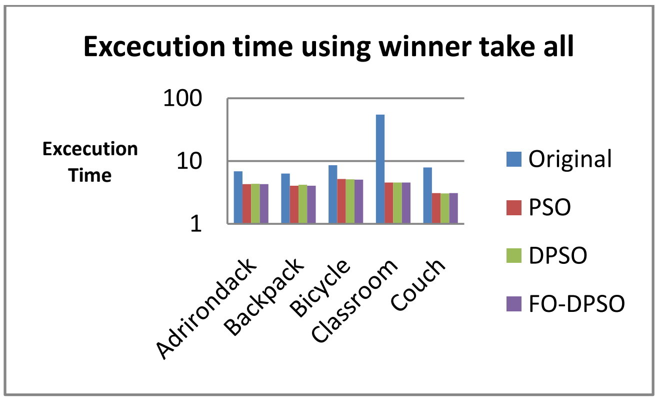


Figure 6. Graph of Execution Time for Winner Take All

Table 4 Execution timeRequired for line growing Algorithm in seconds.

Image name	*Original Disparity*	*PSO*	*DPSO*	*FO-DPSO*
Adirondack	379.498061	177.939	180.6081	140.8252
Backpack	262.46388	87.5464	86.688234	92.01181
Bicycle	509.957358	452.908	441.8667	438.4331
Classroom	4010.955302	126.232	265.73214	190.8848
Couch	448.968969	64.4903	63.633763	68.5349

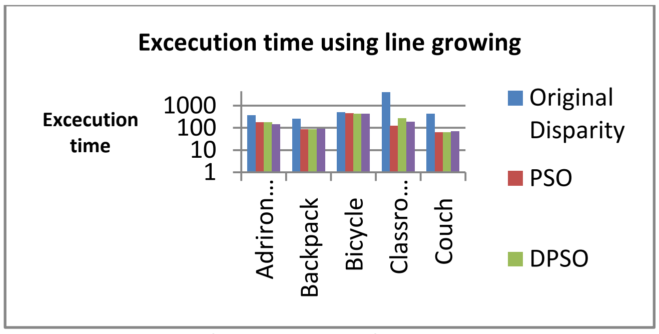


Figure 7 Graph of Execution Time for Line Growing

The values are shown with logarithmic scale on y axis.

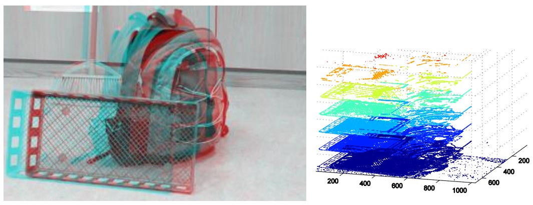
1) *Comparison on the basis of depth levels and 3-D*

Image Generated.

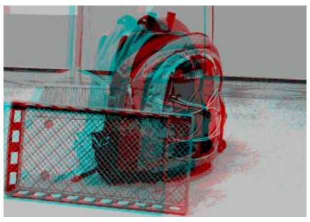
3-D images generated using matching of original views andFO-DPSO segmented views using winner take all algorithms are shown in fig.8a], 8c], .Depth map generated using original stereo matching and FO-DPSO segmented stereo matching are shown in fig.8b] and 8d].

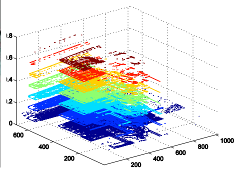
V. Conclusion Future work

Segmentation based stereo matching technique to generate dense disparity map which will be useful in vision systems is designed.3-D image is reconstructed using segmented stereo images. From subjective analysis it is found that the reconstructed 3D images are at par with the original.Reconstructed 3D images are analyzed based on compression ratio. All the three segmentation techniques give 50% compression in 3-D image maintaining subjective image quality. Depth levels obtained from segmented views are comparable with depth levels obtained from original.



a] b]





c] **d]**

Figure 8 Generation of 3-D View

References

[1] Richard Szeliski, "Computer Vision: Algorithms and Applications, "Springer, 2010.

[2] D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nesic, X. Wang, and P. Westling. High-resolution stereo datasets with subpixel-accurate ground truth in German Conference on Pattern Recognition (GCPR 2014), Münster, Germany, September 2014.

[3] http://vision.middlebury.edu/stereo/data/2014/

[4] ArtiKhaparde, ApurvaNaik,ManiniDeshpande, SakshiKhar, KshitijaPandhari, and MayuraShewale, "Performance Analysis of Stereo Matching Using Segmentation Based Disparity Map", ICDT 2013:TheEighthInternational Conference on Digital Telecommunications, 21-26, April 2013, Venice, Italy, pp. 38-43.

[5] Y. Ohta, T. Kanade, "**Stereo by intra- and inter-scanline search using dynamic Programming"**, IEEE Transactions on Pattern Analysis and Machine Intelligence vol.7, 139–154, 1985.

[6] D.B.Fogel "Evolutionary Computation:Toward a new philosophy of machine intelligence, second edition, Piscataway, NJ,:IEEE Press,2000.

[7] J. Tillett et al., "Darwinian particle swarm optimization," in Proc. 2nd Indian Int. Conf. Artif. Intell., 2005, pp. 1474–1487.

[8] Y.Kao,E.Zahara,"A hybridized approach to data clustering"Expert Systems with applications,34(2008),1754-1762.

[9] R.V.Kulkarni,G.K.Venayagmoorthy,Bio-inspired algorithms for Autonomous Deployment and localization of sensor nodes,SMC-C(40),No.6,pp.663-675,Nonember 2010.

[10] P. Ghamisi,M. S. Couceiro, J. A. Benediktsson, and N.M. F. Ferreira, "An efficient method for segmentation of images based on fractional calculus and natural selection," *Expert Syst. Appl.*, vol. 39, no. 16, pp. 12 407–12 417, Nov. 2012.

[11] M. S. Couceiro, N. M. F. Ferreira, and J. A. T.Machado, "Fractional order Darwinian particle swarm optimization," in Proc. Symp. FSS, Coimbra,Portugal, Nov. 4–5, 2011.

[12] T. Bait, D. Boudaoud, B. Matsuzewski, and L.Shark"An efficient Feature Based Matching Algorithm for Stereo Images" Modelling and Imaging, 2006, Page(s):195-202

[13] PedramGhamisi, Micael S. Couceiro, JónAtliBenediktsson, andNuno M.F. Ferreira, "An Efficient Method for segmentation of Images based on Fractional Calculus and Natural Selection", Expert Systems with Applications: An International Journal, vol. 39, iss. 16, November 2012, pp. 1207-1217.

[14] http://www.vision.caltech.edu/bouguetj/calib_doc/ [retrieved:December 2nd, 2013]

[15] www.vision.middlebury.edu [retrieved: August 15, 2009].