Journal of Computers Vol. 29 No. 4, 2018, pp. 110-121
doi:10.3966/199115992018082904009

Research on Improvement of Stereo Matching Algorithm Based on ELAS

Xinyu Hu*, Yuan Wu, Daode Zhang, Lei Qian, Liangyi Wu

College of Mechanical Engineering, Hubei University of Technology, Wuhan, China
huxy_2005@mail.hbut.edu.cn

Received 27 February 2017; Revised 24 July 2017; Accepted 9 August 2017

Abstract. Binocular stereo matching algorithm is real-time and accurate in the binocular vision servo control system, which is very important for the accurate positioning and control of the robot. Based on the study of some advanced stereo matching algorithms, this paper introduces an algorithm of local stereo matching - ELAS (Efficient Large-scale Stereo) algorithm. This local stereo matching is based on processing of the parallax plane. The algorithm overcomes the shortcoming of these current mainstream algorithms that only by setting the maximum disparity value range, can we obtain the best disparity maps. In consideration of processing time and processing effect, this algorithm has best performance. The principle of ELAS algorithm is analyzed thoroughly in this paper. The defects are also put out. Aiming at the problem that the ELAS algorithm does not work well, the improved ELAS algorithm is studied according to the principle of disparity continuity. The disparity map is processed using the guided filter. Experiments show that the improved ELAS algorithm improves the performance of the original algorithm significantly. Without considering the real-time performance, combined with the above-mentioned filters, the processing effect can be close to the real disparity map, which is of great significance in 3D reconstruction and visual servoing.

Keywords: ELAS, real time, stereo matching, wave filtering

1 Introduction

In visual servo system based on binocular vision, the real-time performance of the visual image acquisition and processing is very important [1]. The recognition effect of the visual image processing algorithm affects the accuracy of the motion of the robot. Stereo matching is one of the most important steps to solve the disparity. According to the disparity value and the pixel position, the 3D coordinate information of the target is calculated, and then, controlling robot or manipulator reach the target position, which constitute the basic process of visual servo system. Stereo matching is the process of obtaining the disparity map from the left and right images to the scene in the image. The existing stereo matching algorithms can be divided into two kinds of global and local methods [1].The global method [2-6] can obtain the matching result by minimizing the energy function, the accuracy is higher, but the efficiency is lower, the running time is too long, which cannot meet the real-time application. The local method [7-9] can make use of the neighborhood information in the window to match the single pixel, and the disparity computation depends on the neighboring pixels around the matching points. Stereo algorithms based on local correspondences [10-11] are typically fast, but require an adequate choice of window size. It is very sensitive to the local ambiguity region such as occlusion or texture, and the error matching rate is high. Stereo matching is divided into four steps: matching cost computation, matching cost aggregation, disparity acquisition, disparity refinement. The performance of the algorithm is mainly reflected in the first two steps, the similarity function is calculated by matching the price generally uses the absolute intensity differences) and squared square difference intensity differences etc. On the matching cost aggregation stage, the local stereo matching algorithm is generally added to the window related

* Corresponding Author

smoothing term after the aggregate function, and the global stereo matching algorithm is added to the global constraint. However, the existing stereo matching algorithms can hardly meet the requirements of real time and accuracy based on binocular visual servo. And most algorithms that meet the scene's application have poor real-time performance. With these algorithms, matching a picture of 1 million pixels often requires more than 1s of time. For the algorithms that have great real-time performance, such as SAD algorithm, the rate of mismatch is too high to identify the target location accurately from the disparity maps. Therefore, under the condition of satisfying the requirement of real time, we should minimize the mismatch rate by comparing the most advanced stereo matching algorithms.

The famous open source library (opencv) in the current release of the stereo matching algorithm, selects three advanced algorithms, respectively, Block Matching, Semi-GloBal Matching and Graph Cut. After the optimization of the opencv, the three algorithms can achieve satisfactory results in the effect and speed compared with the original algorithm. Block Matching is a local stereo matching algorithm. It uses a block matching idea, that is, using the local window of the surrounding pixels, to compare the similarity of the two local windows, to obtain the correct parallax. This idea is widely used in some other stereo matching algorithms. Semi-GloBal Matching is also a local stereo matching algorithm proposed by Heiko Hirschm et al. [12] in 2008, which adds the idea of dynamic programming not only to the parallax search on the polar line, but also to find the best parallax value by multi-directional path search, which is called Semi-GloBal. Graph Cut is a global stereo matching algorithm proposed by Martull, S and Martorell, MP and others [13] in 2012 ICPR meeting, based on the graph cut, using the minimum segmentation maximum flow technology, adding global energy constraints, can obtain better disparity image. Geiger et al. Proposed an algorithm at the 2010 ICCV conference - ELAS (Efficient Large-scale Stereo) [14].ELAS algorithm is a fast stereo matching algorithm based on support points. Because of the support point model, it is not necessary to carry out intensive reconstruction, only disparity estimation, greatly reducing the matching time, can calculate 1 million disparity values per second in single-core cpu (i5) [14], is an advanced local stereo matching algorithm. In the Middlebury website published algorithms, the algorithm is ranked second in time / Mp (time / megapixel), and can be applied to visual servo fields with high real-time requirements.

The related works of this paper can be summarized as followed: (1) Comparing the ELAS algorithm with other current advanced stereo matching algorithms to ascertain that the ELAS algorithm can meet our requirements. (2) The working principle of ELAS is introduced in this paper. We make an improvement on grayscale approximation of the ELAS algorithm to improve the accuracy of disparity estimation. In terms of the situation that appears parallax value of occlusion area greater than 0, we ameliorate the algorithm to reduce error point in parallax calculation. (3) We introduce the guided filter to refine a disparity maps. The disparity maps are better, but the real-time performance is very poor. We can appropriate reduce the size of the parallax maps to make the algorithm meet the requirements of real time.

2 Comparison of Common Stereo Matching Algorithm and ELAS

In this paper, all experiments used cpu CORE i7 4710m. The following experiments use the well-known Tsukuba (384×288), Teddy (450×375) and Cone (450×375) test charts on the Middlebury test website. The experiment's false match rate is obtained by comparison with the ground-truth. The parameters of the ELAS algorithm are: $\gamma=5$, $\beta=0.02$, $\sigma=1$ (see the formula below). The remaining parameters such as the parallax range and the SAD window size are the same as the other three algorithms (BM algorithm, SGBM algorithm, GC algorithm), since the maximum parallax value used in the other three algorithms affects the parallax search range and the number of similar regions. So its value on these algorithms have a great impact on the speed and effect., and ELAS does not need to set the value can be obtained with only a parameter of the parallax map, so the value calculated by the ELAS and then set it on the other three algorithms (see Table 1).

Table 1. Comparison of stereo matching algorithms

Stereo matching algorithm	Tsukuba		Teddy		Cone	
	time (s)	False match rate (%)	time (s)	False match rate (%)	time (s)	False match rate (%)
BM algorithm	0.028	45.25	0.035	36.36	0.058	31.33
SGBM algorithm	0.118	35.46	0.275	24.25	0.278	21.55
GC algorithm	3.04	24.54	24.095	34.84	22.945	24.41
ELAS algorithm	0.087	34.69	0.134	25.79	0.132	27.44

Experimental results show that: (1) BM algorithm is the fastest, can be applied to real-time high occasions, but the error matching rate (threshold is a pixel difference, that bad1.0) is too high. (2) From the data point of view, GC algorithm for the longest time, only in dealing with the background is dark, and the background and prospects of a large color difference Tsukuba, have a better performance. When the adjacent patches are large, the parallax judgment is inaccurate, but the result of the algorithm is compared with the ground-truth. The results of the GC algorithm show the most details and clear contours. The items in the scene are the most complete. (3) ELAS algorithm and opencv optimized SGBM algorithm is equivalent to the two algorithms in time and effect are better performance, in dealing with small size Tsukuba diagram, the time is about 0.1s,that is the minimum requirement for real-time processing of frames, but the ELAS algorithm is slightly faster. (4) In general, ELAS algorithm in the effect and time on the best performance, but the error matching rate is still high. (see Fig. 1)

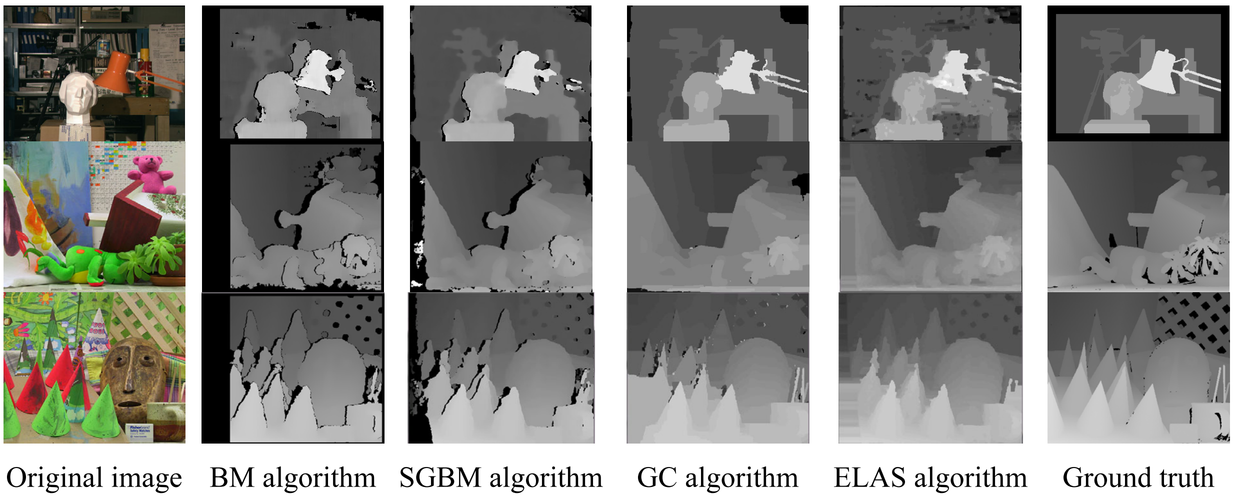


Fig. 1. The result of test graphs after the each algorithm

3 ELAS Algorithm Process and Analysis

3.1 Process of ELAS Algorithm

The algorithm is applied to two images that have been corrected and the left image is the reference image. It calculates the parallax plane through the support points and the Delaunay triangulation, then calculates the parallax value on the parallax plane for the points on the image, sets the threshold value, and selects the best parallax value. The algorithm increases the regional distribution model of observation points, and makes the obtained parallax effect better. The key of the algorithm is to find the support points and calculate the minimum energy optimization terms. The main steps of ELAS algorithm shown in Fig. 2.

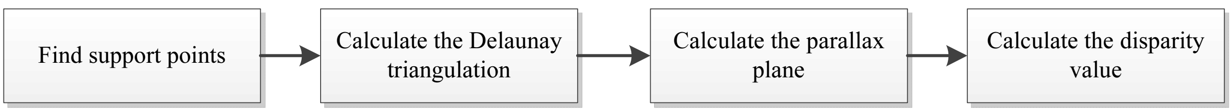


Fig. 2. The main steps of ELAS algorithm

3.1.1 Find Support Points

While a variety of methods for obtaining stable correspondences are available [15-18], we find that matching support points on a regular grid between vectors formed by concatenating the horizontal and vertical Sobel filter responses of 9×9 pixel windows to be both efficient and effective. In ELAS algorithm, the original image is filtered by the 3×3 sobel operator, and the horizontal and vertical response values of each pixel are obtained. Then, for each pixel, the 12 values are selected from the horizontal direction and the 4 value is selected in the vertical direction from the sobel response value of the 5×5 neighborhood around the pixel point form a feature vector $S_{(i,j)} = (S_0\ S_1\ S_2 \ldots S_{15})$. For each pixel((u,v)) on the left image, let the corresponding eigenvector be $S^l{}_{(u,v)}$. The similarity(Equation (1)) is calculated according to the different parallax values using the eigenvectors ($S^r{}_{(u-d-2,v-2)}$, $S^r{}_{(u-d+2,v-2)}$, $S^r{}_{(u-d-2,v+2)}$, $E(s) = \sum \left| S^l_{(x,y)} - S^r_{(x-d_1,y)} \right| x = \begin{cases} u-2 \\ u+2 \end{cases} y = \begin{cases} v-2 \\ v+2 \end{cases}$) of the four vertices in the 5×5 neighborhood of the pixel corresponding to the right image, If the ratio of the minimum and minimum values of E(s) is within the set threshold, the parallax value is taken as the parallax value of the candidate support point.

$$E(s) = \sum \left| S^l_{(x,y)} - S^r_{(x-d_1,y)} \right| \qquad x = \begin{cases} u-2 \\ u+2 \end{cases} \quad y = \begin{cases} v-2 \\ v+2 \end{cases} \tag{1}$$

$$E(s) = \sum \left| S^r_{(x,y)} - S^l_{(x+d_1,y)} \right| \qquad x = \begin{cases} u-d-2 \\ u-d+2 \end{cases} \quad y = \begin{cases} v-2 \\ v+2 \end{cases} \tag{2}$$

The algorithm uses a priori and posteriori comparison in order to obtain a better support point. That is, the left image is used as a reference, and the parallax value d is obtained using Equation (1). And then through using the right image as a reference, the use of Equation (2) to obtain the disparity value d_2 .When $\mathrm{d}_2 \geq 0$, if $|\mathrm{d}_2 - \mathrm{d}| < \mathrm{Threshold}$, the point is selected as the support point, thus obtaining a reliable point in the pixel parallax coordinate system $(\mathrm{u,v,d})$.

3.1.2 Optimization Model

The original image is divided into meshes (20 by 20 in the original text) to determine the number of points in each grid. For each reference point, find all the support points in the 3×3 region centered on the corresponding grid, and obtain some corresponding points by using the parallax values of these support points as the alternative parallax of the reference point. The similarity between these points and the reference point is calculated, and the point that minimizes the value of the similarity function (formula (3)) is selected. In this way, the best support point, alternative parallax and the minimum similarity function value are obtained. Let the parallax change within the allowable range, adding optimization item (formula (5)) and calculate the value of the similarity function after the optimization. if the minimum value of the similarity function is less than the values obtained before the substitution of alternative disparity values, otherwise unchanged. Thus the optimal disparity values are obtained. In order to obtain the search range of parallax, we use the support point to do Delaunay triangulation, and then the parallax plane function (equation (4)) of the triangle is established according to the three points of the triangle, and the parallax plane equation is solved. Pixel coordinates into, to solve the corresponding parallax value d_{plane}, and then add the deviation in the value d_u, thus the range of parallax to be selected, this is $d_{plane} - d_u \leq d \leq d_{plane} + d_u$.

$$E(s) = \sum \left| S^l_{(x,y)} - S^r_{(x-d_1,y)} \right| \quad x = u \quad y = v \tag{3}$$

$$E(d) = \beta \times E(k) - \log\left[\gamma + \exp\left(-\frac{\left[d - \mu(S, o^{(l)})\right]^2}{2\sigma^2}\right)\right] \quad 0 \le i \le 3 \tag{4}$$

In Fig. 3, the disparity is calculated by inputting the reference point, the support point and the parameter (γ, σ), and then the corresponding points are obtained by the disparity, the reference point and the parameter $w = \frac{\left\{\log\gamma - \log\left[\gamma + \exp\left(-\frac{\left[d - \mu(S, o^{(l)})\right]^2}{2\sigma^2}\right)\right]\right\}}{\beta}$. According to [1], the optimization term is:

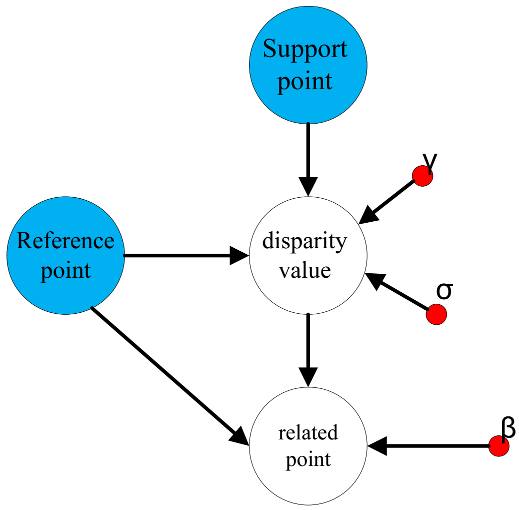


Fig. 3. Parallax calculation model

$$E(d) = \beta \times E(k) - \log\left[\gamma + \exp\left(-\frac{\left[d - \mu(S, o^{(l)})\right]^2}{2\sigma^2}\right)\right] \tag{5}$$

Including: d is the current disparity value;

$|d - d_k|$ is the Gaussian distribution of the reference point, the general take 0;

In order to be better used for program calculation, conversion formula (5):

$$0 \le d \le width - u \tag{6}$$

The general disparity of the similarity function formula is:

$$E(k) = \sum \left| S^l_{(x,y)} - S^r_{(x-d_1,y)} \right| \quad x = u \quad y = v \tag{7}$$

That is, for a certain pixel, the similarity value of the value of two. (1) The parallax is estimated by the parallax near the parallax, and the similarity value is calculated using equation (6). (2) Other parallax values, using the formula (7) to calculate the similarity value function value. Finally, the final parallax is obtained by using the WTK algorithm, that is, the parallax with the smallest similarity function is chosen as the final parallax of the pixel.

3.2 Analysis of ELAS Algorithm

Let $w = \frac{\left\{ \log\gamma - \log\left[\gamma + \exp\left(-\frac{\left[d - \mu(S, o^{(l)}) \right]^2}{2\sigma^2} \right) \right] \right\}}{\beta}$. ELAS uses SAD (absolute value aggregation) to calculate similarity, and the smaller the similarity function value indicates the more similar the region. The smoothing term $Threshold\_d$ is added after the cost aggregate function, and the similarity function value of the parallax value (set) is estimated by the parallax plane. The function curve plotted with the parameters used in this article (the abscissa is $|d - d_k|$, where $0 \le d \le width - u$, the ordinate is w). As shown in Fig. 3, the value of the similarity function $E'(d)$ is reduced by 9 at the estimated parallax value (the abscissa is zero). By carefully analyzing the process of the original algorithm, in fact, the final parallax of a pixel is chosen only in the alternative parallax and estimated parallax of the parallax corresponding parallax plane. The original algorithm simply adds the distribution function of the parallax selection, and does not take into account the situation around the pixel, so there will be parallax error. But also for the occlusion of certain parallax discrimination will be a logical error, that is, some of the occlusion in the occlusion area although the calculation is correct, but because the background has a larger parallax, the prospects have smaller parallax. During the subsequent padding area, the prospect of a smaller parallax value is selected. This makes the original area of the background is judged as the foreground, resulting in a logical error. But ELAS creatively introduced the bridge of the computer graphics and computer vision - Delaunay triangulation, and used the concept of parallax plane, the parallax estimation, greatly reducing the algorithm time. It is a great practical value of the advanced stereo matching algorithm.

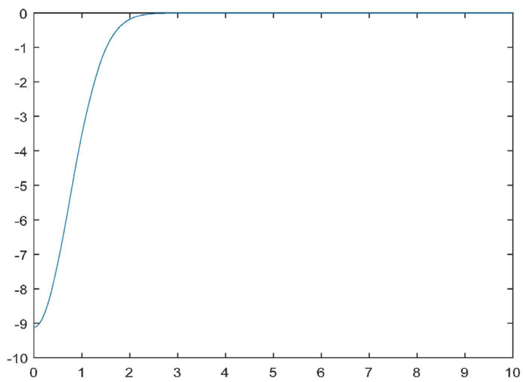


Fig. 4. Curve of smoothing function w

4 Principle of Improved ELAS Algorithm

The LS-ELAS algorithm is different from the famous ELAS algorithm based on line segmentation [19]. The LS-ELAS algorithm has been improved on the basis of ELAS algorithm. The principle of the LS-ELAS algorithm is to extract support points that contain discontinuous data of depth by extracting image edges and add the length data of the edges to the original optimization function. The LS-ELAS algorithm can achieve higher precision rate in sparse image sets. But our improved ELAS algorithm is adopted the parallax continuity principle. This way can have good reinforcement effect of the original algorithm. This improved ELAS can be both applied to dense and sparse image sets. And there are two kinds of statements for the principle: (1) In the scene (in the same image), the adjacent gray value of the same place parallax continuous, that is, no gray-scale mutation in the region, parallax is continuous. This is a good understanding of the scene which is the same object or background parallax is continuous, no parallax mutation. Parallax mutations indicate a new prospect or background. (2) In the occlusion area, the parallax parallax value of the adjacent gray scale value in the same image is not continuous. This is actually the reverse understanding of the first expression, which is set in the occlusion area, because in the occlusion area, usually the foreground occlusion background, or background occlusion of the situation, then, the emergence of new prospects or Background (that is, the occlusion area appears parallax value greater than 0 area), there must be gray-scale mutation. According to the above expression 1, taking into account a pixel in the left camera image, if the gray value of its left adjacent pixel is equal to its gray value, there is no edge, indicating that their parallax is continuous. In the process of selecting the parallax (second choice), the final parallax is the choice of the parallax of the corresponding parallax plane or the parallax in the process of estimating the parallax which is close to the left parallax value. This principle can be used to select more accurate parallax. However, it is necessary to set a threshold to indicate that the parallax is continuous, and the improved method is used when the parallax change value is less than the threshold. Let the threshold be $Threshold\_d$, the candidate parallax at this point is set to d_{out} , the corresponding similarity function value is E_val_{out} , the estimated parallax at that point is d_{inner} , the corresponding similarity function value is E_val_{inner} , The parallax value of the adjacent pixel (equal to the gray value of the point) is d_{left} , and the final parallax is d .

$$d \equiv \begin{cases} equal(d) & \left|d_{inner} - d_{left}\right| \leq Threshold\_d \quad or \quad \left|d_{\text{out}} - d_{left}\right| \leq Threshold\_d \\ non(d) & else \end{cases} \tag{8}$$

Among them:

$$equal(d) = \begin{cases} d_{out} & \left|d_{inner} - d_{left}\right| > \left|d_{out} - d_{left}\right| \\ d_{inner} & \left|d_{inner} - d_{left}\right| \leq \left|d_{out} - d_{left}\right| \end{cases} \tag{9}$$

$$non(d) = \begin{cases} d_{inner} & E\_val_{inner} \leq E\_val_{out} \\ d_{out} & E\_val_{inner} > E\_val_{out} \end{cases} \tag{10}$$

After calculating the final parallax of all the pixels, after obtaining the initial disparity map, according to the expression 2, in order to reduce the parabax of the logical error in the occlusion area, the subsequent processing can remove the error areas. Consider finding the gray point in the area where there is a logical error in the occlusion area. The parallax of the masked area that has been processed by the algorithm has been set to a negative value. The basic steps are: 1. Identify the areas where there may be a logical error surrounded by the occlusion area, marking the pixels in these areas. Provided that the parallax value in the area marked by the process is continuous.2. Scanning these pixels from left to right, from the starting pixel point, that is, adjacent to the left side of the parallax value is negative, to determine whether the gray-scale mutation. If it is not a mutation point, it will set its parallax value to a negative number, if it is a sudden point, mark the mutation mark = -1.3. Move to the right if the adjacent left pixel is a marker mutation point, or if its adjacent left pixel is not a marker burst and the point parallax is not negative. It is judged whether the gray level of the current pixel is abruptly changed, and if it is not a sudden point, the current parallax is maintained, and if it is a sudden point, its parallax is set to a negative value and the mutation is marked. If the adjacent left pixel is not the marker mutation point, and the point parallax is negative, it is judged whether the gray level of the current pixel is abrupt. If it is not a mutation point, its parallax is set to a negative value and marks the mutation, and if it is a sudden point, the current parallax is maintained. 4. Repeat 3 until the line is scanned and move to the next line to repeat steps 2 and 3 until all marker areas have been scanned.

The key to the above process is to mark out areas where logical errors may occur and to find gray-scale mutations. In order to mark these areas, all connected domains can be found, and parallax factors are added on the basis of the original connected domain segmentation, and when the parallax is continuous in the region, it is selected as a separate connected domain. Since the concatenation domain is surrounded by a point where the gray scale value is a negative or gray-scale mutation, this is consistent with the characteristics of a region where a logical error may occur, and in order to remove a wide range of backgrounds and prospects, these regions can be identified by limiting the size of the connected domain. Let the limit threshold be $Threshold\_s$. It is relatively simple to find the gray point mutation point, and only if the absolute value of the gray scale difference between the current gray level and the adjacent pixel point on the left side is greater than the limit threshold, this threshold is set $Threshold\_g$. Because the gray value of the noise and the gray value of the surrounding pixels are very different, the gray point of the gray point is easy to be affected by the noise, which will lead to some errors.

In order to eliminate the noise need to add step processing, but in order not to affect the real gray point mutation point in the processing area cannot use a simple median filter. Sobel's features also describe these true gray-scale mutations, and the original algorithm is also calculated (looking for support points in the process), without additional computational burden, so the following approach: For the processing of a pixel, calculate its 3×3 window gray mean value $mean1$, and then remove the pixel gray value of the average value $mean2$. If $|mean2 - mean1| \times 8 > Threshold\_g$, it is judged whether the sobel horizontal response value of the pixel or the absolute value of the vertical response value is greater than the threshold value $Threshold\_f$. If it is greater than, indicating that the point is not a noise; if less than, indicating that the point is noise, you need to use $a_k = \frac{1}{|w|}\sum_{i \in w_k}\left(\frac{(I_i \times P_i - \mu_k \times \overline{P}_i}{\sigma^2 + \varepsilon}\right)$ instead of the gray value.

5 Comparison between Improved ELAS Algorithm and Original Algorithm

The Middlebury 2014 datasets were used as the test dataset, and the disparity maps generated by the original algorithm and improved algorithm were uploaded to Middlebury website to get the test results. In the Improved algorithm, $b_k = P_k - a_k \times \mu_k$, $q_i = \frac{1}{|\omega|}\sum_{k:i \in \omega}\left(a_k \times I_i + b_k\right) = \overline{a}_i \times I_i + \overline{b}_i$, $Threshold\_g = 10$, $Threshold\_f = 10$.

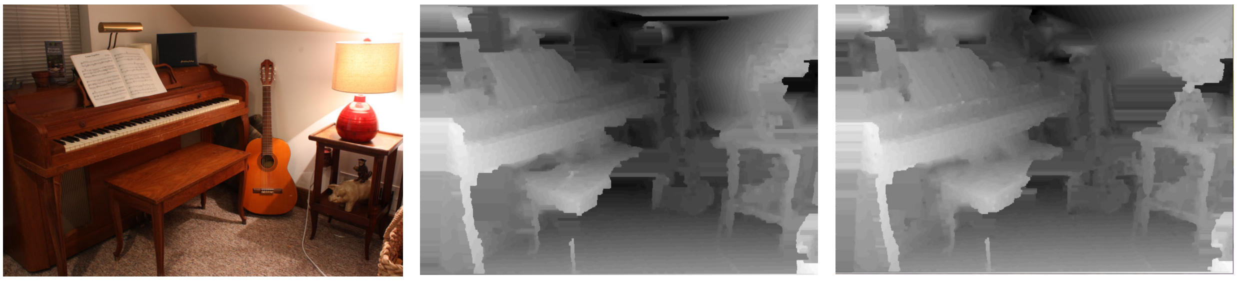
The experimental results show that the error matching rate of these tests is reduced. In the Adirondack test chart, the improved algorithm reduces the false matching rate by 38.7%, the time increases by 26ms, the 38% decrease in the Playtable, and the time increase of 25ms. The minimum false match rate on the PlaytableP test chart is 15.63% and the time is increased by 16ms. It can be seen that the improved algorithm greatly reduces the false match rate, and because the original algorithm has marked the logical error may occur in the region, and calculate the sobel response value, so do not need a lot of additional calculations to increase the run time. Only need to judge and assignments can be, and because of the pixels in these areas less, so the increase in time is not much, an average of about 20ms. In the experiment described in Table, for processing dense data sets, the effect of using our improved algorithm is better than the LS-ELAS algorithm. And with this improved algorithm, more than 1 million pixels can be processed in a second. Therefore, the improved algorithm also has real-time application value. Owing to the improvement aimed at calculating process of ELAS, the improvement has impact only on the ELAS algorithm.

6 Refine the Outline of the Object in the Disparity Map

The improved algorithm reduces the mismatch rate and enriches some details of the original algorithm, which is illustrated by the Piano-L test chart with high mismatch rate in Table 2, as shown in the Fig. 5.

Table 2. Nonocc of the original algorithm and the improved algorithm are at bad 2.0

	Adirondack	ArtL	Jadeplant	Motorcycle	MotorcycleE	Piano	PianoL	Pipes	Playroom	Playtable	PlaytableP	Recycle	Shelves	Teddy	Vintage
ELAS	27.1	25.9	49.3	23.8	24.0	31.4	46.6	27.2	43.8	55.4	*24.3*	30.5	52.4	18.7	45.1
Improved ELAS	16.6	20.7	35.0	16.8	16.1	22.0	33.2	22.4	32.7	34.1	*20.5*	19.7	43.2	14.5	36.3
LS_ELAS	34.5	10.6	40.6	21.4	23.0	33.1	48.8	23.0	38.5	51.7	21.1	36.2	54.6	14.4	50.2
ELAS time(s)	0.4	0.088	0.31	0.331	0.361	0.297	0.379	0.322	0.299	0.317	0.335	0.302	0.398	0.134	0.321
Improved time(s)	0.426	0.113	0.327	0.342	0.383	0.312	0.395	0.339	0.313	0.342	0.351	0.32	0.417	0.151	0.329
LS_ELAS time	0.412	0.12	0.295	0.33	0.36	0.306	0.382	0.327	0.294	0.315	0.33	0.309	0.402	0.141	0.324



Original image Original algorithm Improved algorithm

Fig. 5. The result of PianoL after the original algorithm and the improved algorithm

Table lamp in the original algorithm is not shown, but in the improved algorithm can be very clear to see. However, in some cases where the detail requirements are high, it needs a disparity map similar to the true parallax. The details provided by the improved algorithm do not recognize the outline or other characteristics of the object better. Since the gray scale distribution feature in the original image can better describe the contour feature. It is considered to use the original image as the guided image to guide the filtering. According to the literature [20] on the obtained parallax map using the guide filter, the filter using the idea of linear, guide map and filter with a linear correspondence between the theoretical formula is as follows:

$$a_k = \frac{1}{|w|} \sum_{i \in w_k} \left(\frac{(I_i \times P_i - \mu_k \times \overline{P_i}}{\sigma^2 + \varepsilon} \right) \tag{9}$$

$$b_k = P_k - a_k \times \mu_k \tag{10}$$

$$q_i = \frac{1}{|\omega|} \sum_{k:i \in \omega} \left(a_k \times I_i + b_k \right) = \overline{a}_i \times I_i + \overline{b}_i \tag{11}$$

I is the guide image, P is the Filted image, w_k and ω is the window, μ_k is the window mean, σ is I is the guide image, P is the Filted image, w_k and ω is the window, μ_k is the window mean, σ is the intra-window variance, ε is the regularization item, q is the filter output. Select the left camera image as the guide image. $w = 9$, $\omega = 9$, $\varepsilon = 500$. Using the improved box-filter technique [21], the complexity of the algorithm is changed to $O(1)$ to accelerate the processing and reduce the running time.

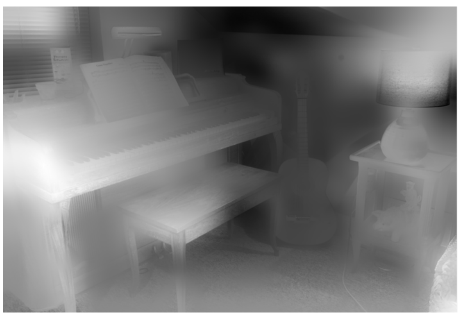
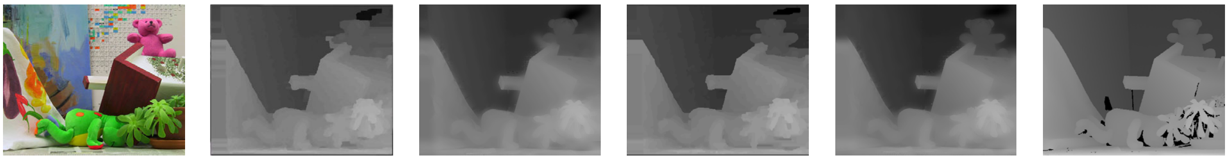


Fig. 6. The result of PianoL after the original algorithm and filtering

By filtering, the new disparity map shows a lot of details of the object in the scene and is very close to the real scene. Teddy test graph to compare the original algorithm and improve the effect of filtering before and after the algorithm. The results of the Teddy test graph are compared with those of the original algorithm and the improved algorithm (see Fig. 7).



Original image | the Original algorithm | after filtering | the improved Algorithm | after Filtering | Ground Truth

Fig. 7. The result of Teddy after the each algorithm and filtering

Experiments show that the improved weighted median can effectively improve the original disparity map in detail. However, the time of filtering Teddy (450×375) is 0.53s, the time of processing PianoL ($\frac{1}{16}$) is 1.04s, the average is about 350,000 pixels / second. Processing time is too long, does not meet the requirements of real-time processing.

7 In Real Time Applications

The algorithm has been optimized above; consider other aspects except optimization algorithm to reduce run time. In general real-time applications, if the algorithm is too slow, cannot meet real-time requirements. Furthermore, the algorithm time is related to the size of the processed image, and the obtained image can be reduced or cropped before being processed to meet the real-time requirement.

The experimental results show that the reduction of the image size can reduce the time of the improved algorithm and the filter algorithm (see Table 3). The ratio of the running time of the ELAS algorithm in different size images is proportional to the ratio of the size. The PianoL image (176×120), which is one-sixteenth in image size, has a total processing time of 0.096s, which can meet the requirements of real-time processing. The final disparity diagram is shown in Fig. 8.

Table 3. Running times at different sizes

Images with different sizes	Time of the original algorithm(s)	Time of the improved algorithm(s)	Time of flinging(s)
PianoL (707×480)	0.379	0.395	1.04
PianoL (353×240)	0.096	0.081	0.34
PianoL (176×120)	0.029	0.032	0.064

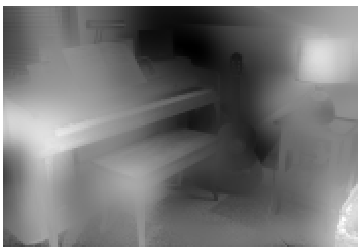


Fig. 8. The result of PianoL ($\frac{1}{16}$) after the original algorithm and filtering

8 Conclusion

Based on the improved parallax continuous algorithm, the matching effect of the original algorithm is greatly improved, and the running time is also increased little, basically can meet the real-time requirements. In real-time applications where contours are required, the parallax can also be improved by reducing the size of the image and using guided filtering.

In addition, the improved algorithm uses only the adjacent gray-level information, in Table 2; some of the test pattern of the mismatch rate is still high. In the following research, we will consider adding more effective information to further reduce the mismatch rate, further explore how to select the parallax value more rationally and accurately, and design more effective similarity function and smoothing function or follow-up processing.

Acknowledgements

This work is supported by Nature Science Foundation of Hubei Province (No. 2016CFB653), the Education Department Project of Hubei Province (No. D20151406), WuHan science and technology support program (No. 2017010201010137)

References

[1] D. Schastein, R. Szeliski, A taxonomy and evaluation of dense two-frame stereo correspondence algorithm, International Journal of Computer Vision 47(2002) 7-42.

[2] Q. Yang, Stereo matching using tree filtering, IEEE Transactions on PAMI 37(4)(2015) 834-846.

[3] J. Lu, H. Yang, D. Min, M.N. Do, Patch match filter: efficient edge-aware filtering meets randomized search for fast correspondence field estimation, IEEE Conference on Computer Vision and Pattern Recognition 9(4)(2013) 1854-1861.

[4] T. Taniai, Y. Matsushita, T. Naemura, Graph cut based continuous stereo matching using locally shared labels, in: Proc. 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[5] F. Besse, C. Rother, A. Fitzgibbon, J. Kautz, PMBP: patch match belief propagation for correspondence field estimation, International Journal of Computer Vision 110(1)(2014) 2-13.

[6] Q. Yang, Stereo matching using epipolar distance transform, IEEE Transactions on Image Processing 21(10)(2012) 4410-4419.

[7] L.D. Stefano, M. Marchionni, S.A. Mattoccia, A fast area-based stereo matching algorithm, Image and Vision Computing 22(12)(2004) 983-1005.

[8] L. De-maeztu, S. Mattoccia, A. Villanueva, R. Cabeza, Linear stereo matching, International Conference on Computer Vision 24(4)(2013) 1708-1715.

[9] Y. Xu, Y. Zhao, M. Ji, Local stereo matching with adaptive shape support window based cost aggregation, Applied Optics 53(29)(2014) 6885-6892.

[10] R. Zabih, D. Scharstein, R. Szeliski, A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, International Journal of Computer Vision 47(1-3)(2002) 7-42.

[11] M. Weber, M. Humenberger, W. Kubinger, A very fast census-based stereo matching implementation on a graphics processing unit, in: Proc. IEEE International Conference on Computer Vision Workshops, 2009.

[12] H. Hirschm, Stereo processing by semiglobal matching and mutual information, IEEE Transactions on Pattern Analysis and Machine Intelligence 30(2)(2007) 328-341.

[13] S. Martull, M. Peris, K. Fukui, Realistic CG stereo image dataset with ground truth disparity maps, Technical Report of Ieice Prmu 111(2012) 117-118.

[14] A. Geiger, M. Roser, R. Urtasun, ELAS: efficient large-scale stereo matching, ACCV 6492(2010) 25-38.

[15] A.F. Bobick, S.S. Intille, Large occlusion stereo, International Journal of Computer Vision 33(1999) 181-200.

[16] X. Hu, P. Mordohai, Evaluation of stereo confidence indoors and outdoors, Computer Vision and Pattern Recognition 238(6)(2010) 1466-1473.

[17] X.H. Zhang, G. Li, C.L. Li, H. Zhang, J. Zhao, Stereo matching algorithm based on 2D delaunay triangulation, Mathematical Problems in Engineering 3(2015) 1-8.

[18] R. Sara, Finding the largest unambiguous component of stereo matching, European Conference on Computer Vision 2352(2002) 900-914.

[19] R. Ait-Jellal, M. Lange, B. Wassermann, A. Schilling, A. Zell, LS-ELAS: line segment based efficient large scale stereo matching, in: Proc. IEEE International Conference on Robotics and Automation, 2017.

[20] K. He, J. Sun, X. Tang, Guided image filtering, European Conference on Computer Vision 35(6)(2010) 1-14.

[21] M.J. Donnell, Box-filtering Techniques, Computer Graphics and Image Processing 17(1)(1981) 65-70.