Chapter 3

Proposed Image Registration Algorithm

3.1 Introduction

A comparison of three algorithms: Zhang, W.-C. Kao, and Ward were presented in chapter 2. Among three algorithms, Ward's algorithm is preferred for its high efficiency. Section 3.2 will describe of the experiments designed to examine the accuracy of Ward's algorithm and the discussions of its drawbacks mentioned before. In section 3.3, based on the results of section 3.2, an improved registration algorithm will be proposed to solve the issues of Ward's algorithm. Section 3.4 will be the summary.

3.2 Experiment Models

We firstly examined the accuracy of Ward's algorithm by making the artificial scene in laboratory. If the result was acceptable, we could demonstrate that our program of Ward's algorithm is reliable. Following will be the examination of real scene outdoors.

3.2.1 Laboratory

(1) Experiments：

To examine Ward's algorithm, a series of images were taken under different positions and different exposed intensities. The schematic drawing of the setup shown in **Fig. 3-1** consisted of a Nikon D70 camera, a laser pointer used to confirm the positions of the taken pictures, and a screen. First, three pictures were taken under -1EV, 0EV, and +1EV for every 5mm-shift of the camera from -30 mm to +30 mm along the x-direction. After that, three pictures were

again taken under -1EV, 0EV, and +1EV for every 5mm-shift of the camera from -30 mm to +30 mm along the y-direction. An example of two images taken under 0EV for 0 mm-shift and +30 mm-shift in x-direction is shown **Fig. 3-2**. The red-point on the grid paper of each image can be as a reference for current position. Another example of two images taken under -1EV and +1EV is shown in **Fig. 3-3**.

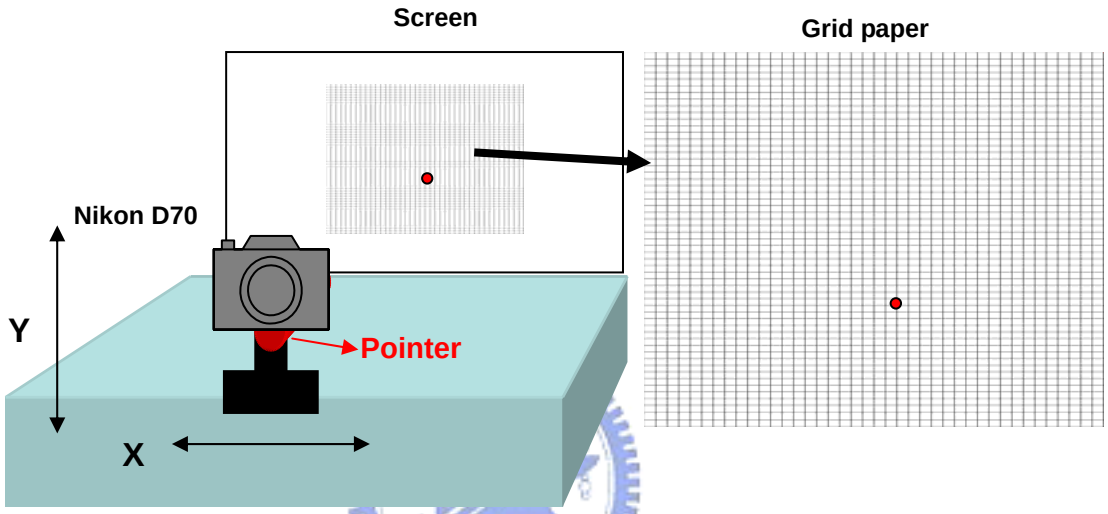
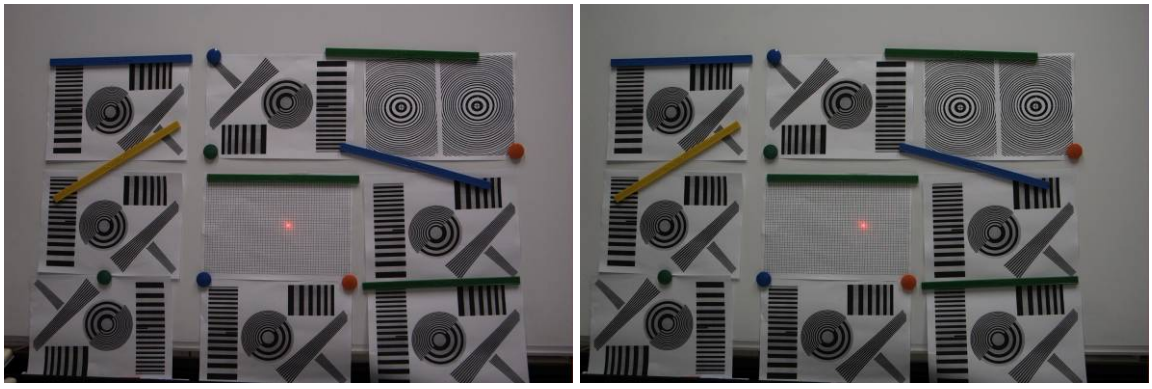
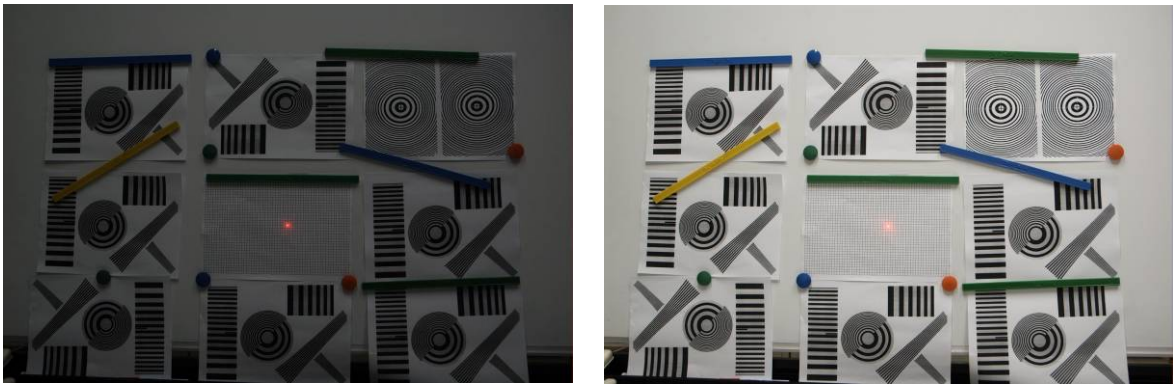


Fig. 3-1 The configuration of the verified model.



(a) **(b)**

Fig. 3-2 Images are taken by different position. (a) The fixed image. (b) The image of 30mm-shift along the +x direction.



(a) **(b)**

Fig. 3-3 Images are with different exposures. (a) -1EV and (b) +1EV.

(1) Results and Discussions：

In the current experiment, thirteen images were taken along the x-direction and the y-direction respectively. Along the x-direction, the relationship between the positions of taken images and the shifted pixels derived from Ward's algorithm is shown in **Fig. 3-4**. There are three curves, which represent the images taken under -1EV, 0EV, and +1EV. These curves are linear, which matches well with the prediction of Ward's algorithm. For the images taken along y-direction, the characteristic relationship shown in F**ig. 3-5** deviates a little from the prediction of Ward's algorithm, which can be attributed to the serious instability in the y-direction during photo shooting time frame. From the results, the linearity in both directions shows the reliability of Ward's algorithm.

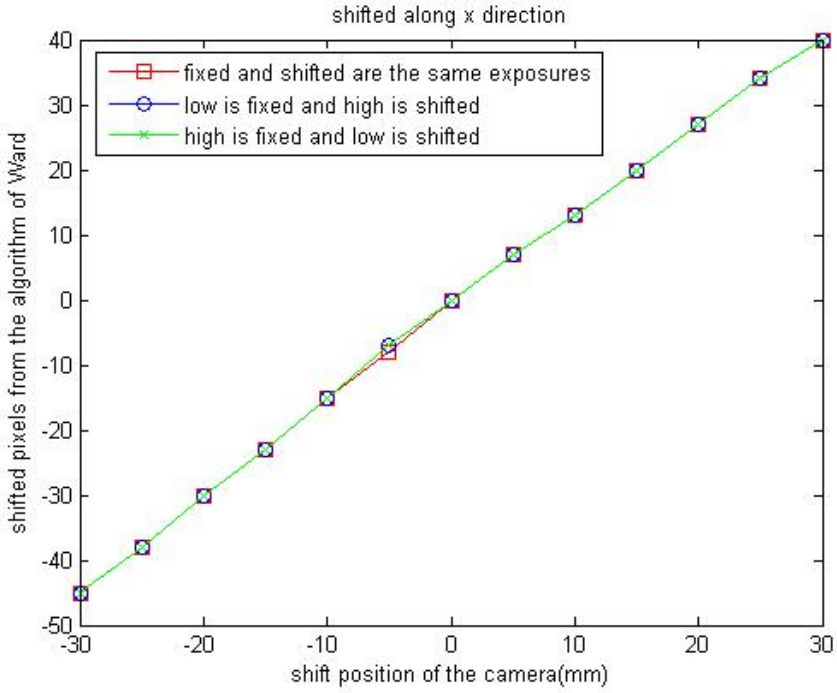


Fig. 3-4 Camera is shifted by x direction. The reference image is fixed and the test images are shifted from -30mm to 30mm.

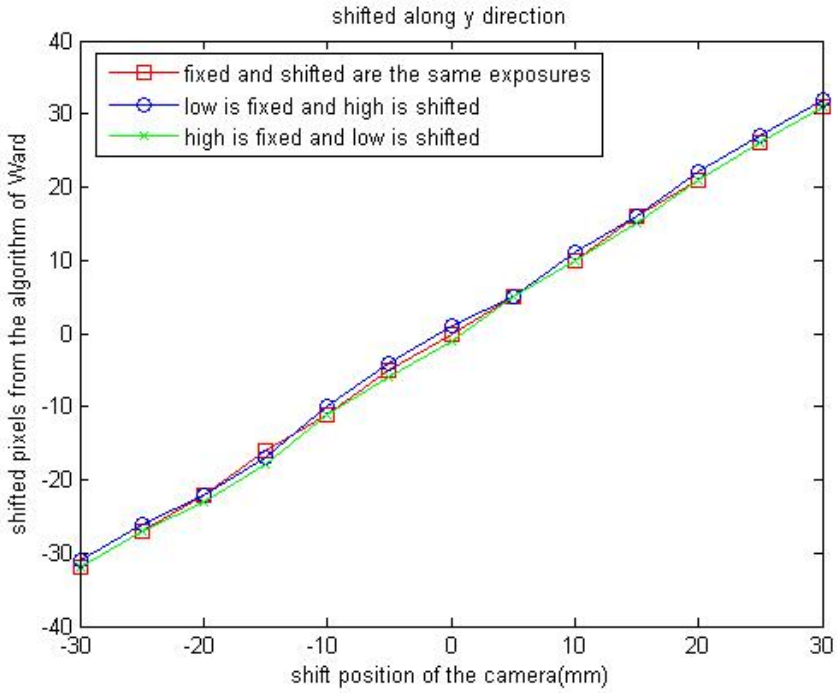


Fig. 3-5 Camera is shifted along y direction. The reference image is fixed and the test images are shifted from -30mm to 30mm.

3.2.2 Outdoors

(1) Experiments：

To verify Ward's algorithm under stricter conditions, a series of images were taken outdoors under different positions. The setup used outdoors was the same as that used in the laboratory. To reduce unexpected errors, the plat form employed to set up the camera had to be flat, so a level was used to confirm the horizon. Limited to the equipments, the camera was only allowed to shift along the x-direction. An example of two images taken under 0EV for 0 mm-shift and +25 mm-shift in the x-direction is shown in **Fig. 3-6**.

(a) **(b)**

Fig. 3-6 Two images are taken outdoors. (a) The fixed image and (b) the image of 25mm-shift along the +x direction.

(2) Results and Discussions：

In the current experiment, thirteen images were taken along the x- direction. The relationship between the positions of taken images and the shifted pixels derived from Ward's algorithm is shown in **Fig. 3-7**. The shifted pixels from the prediction of Ward's algorithm are between -2 and +2. Unlike the results obtained in the laboratory, the curve deviates seriously from linearity. The cause is the amount of shifted pixels varies with the position of the scene being taken. To clarify the problem, each image was divided into two parts: one part is taken from the near scene, such as the girl and the color checker carried by her in **Fig. 3-6**, and the

other part is taken from the far scene, such as the woods in **Fig. 3-6**. Therefore, each image can be characterized according to three conditions: the near scene, the far scene, and the whole scene. The relationship between the positions of taken images under different scenes and the shifted pixels derived from Ward's algorithm is shown in **Fig. 3-8**. These curves demonstrate that the amount of the shifted pixels derived from Ward's algorithm is related to the distance of taken scenes.

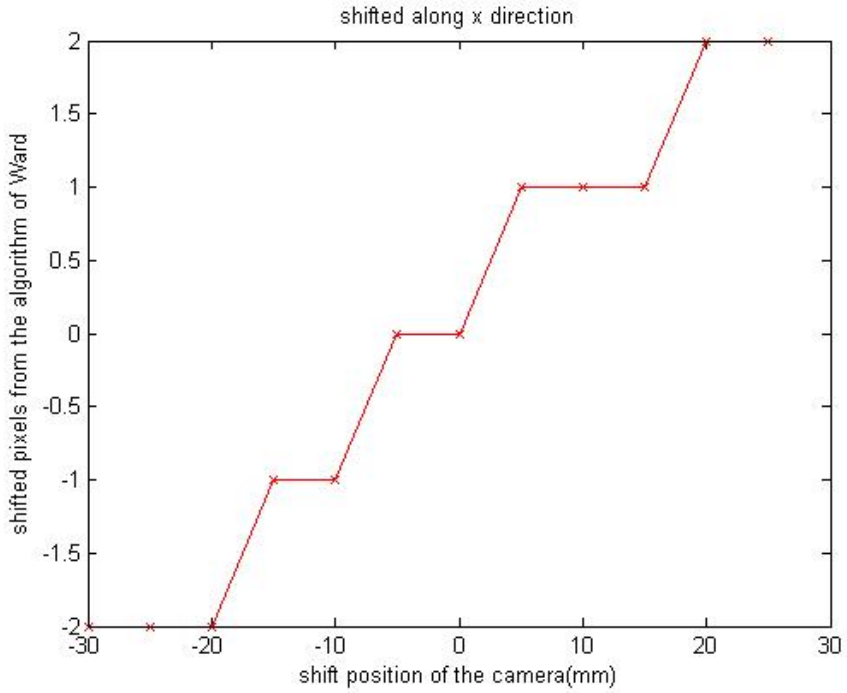


Fig. 3-7 Camera is shifted along the x-direction and the shifted pixels from Ward's results are between -2 and 2.

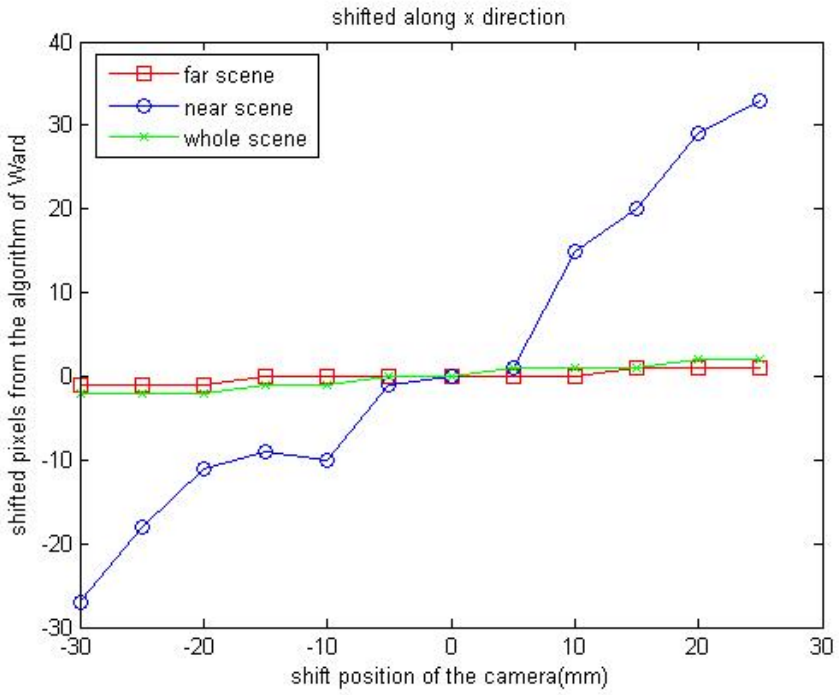


Fig. 3-8 The result is divided into near scene, far scene, and whole scene.

Another example of two images taken under 0EV for 0 mm-shift and +25 mm-shift in the x-direction is shown in **Fig. 3-9**. This time the far scene is closer to the near scene. The above analysis is repeated. The relationship between the positions of taken images under different scenes and the shifted pixels derived from Ward's algorithm is shown in **Fig. 3-10**. Though the background is near, the slope of the near scene is steeper than that of the far scene.

(a) **(b)**

Fig. 3-9 Two of sequential images with another background. (a) The fixed image and (b) the image of 25mm- shift along the +x direction.

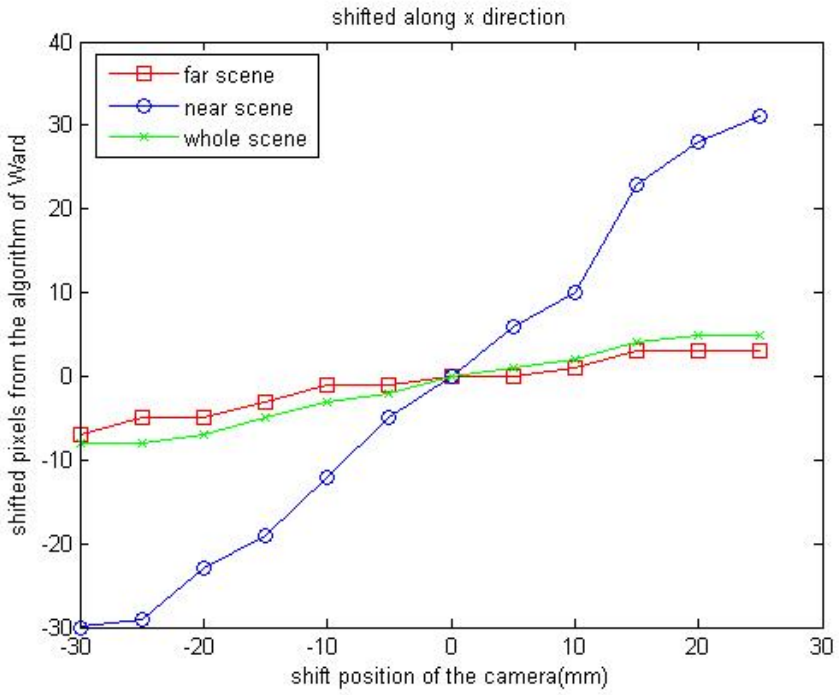


Fig. 3-10 Result is divided into near scene, far scene, and whole scene.

For practical applications, people always prefer to take portrait in front of special scenery. Therefore, the subject of our examined scenes involves both the portrait and the scenery. According to above results, we conclude that the shifted pixels are much different for near and far scenes. As a result, the global motion vector is unreliable.

3.3 Proposed Model

3.3.1 The issues of Ward's algorithm

The issues of Ward's algorithm mentioned in chapter 2 are listed in **Table 3-1**. In addition to good efficiency, we analyze other issues as follows:

Table 3-1 The list of issues, their causes, and improved methods.

	Efficiency	Memory buffer	Complexity (multiplication)	For different exposures	Adaptation for near/far scene
Ward	**O**	**×**	**Δ**	**O**	**×**
Cause	**Binary map**	**Process entire image**	**Pyramid decomposition**	**Medium value (histogram)**	**Global motion vector**
Improvement	**Maintain**	**Cut image**	**No decomposition**	**Mean value (average)**	**Local motion vector**

(1) Memory buffer and adaptable for near/far scene：

Processing image with entire size costs huge memory buffer. Cutting image with macro block and then processing each block once is an improved method. In addition, in order to separate the difference between near and far scenes, cutting image makes them distinguishable by processing each block, which has its own motion vector. The method we use to evaluate the motion vector of each block is called local motion vector (LMV).

(2) The issue of complexity：

Filtering an image requires huge steps of multiplication. As aforementioned discussion of Ward's algorithm, filtering is required because of pyramid decomposition. Therefore, cutting an image into macro blocks is a good substitution for the complicated multiplication.

(3) For different exposures：

Although medium threshold bitmap (MTB) can achieve similar patterns for different exposed images, finding histogram required extra memory buffer. In our experiments, the pattern of mean threshold bitmap is similar to the pattern of medium threshold bitmap, but the

requirement of extra memory buffer is almost zero. As a result, mean threshold bitmap is a good choice.

3.3.2 Our Proposed Algorithm

Our proposed algorithm processed sensor raw image, which was directly captured from CCD/CMOS sensor, thus R, G, and B channels of this image are obtained individually. In our algorithm, we process the gray-level image (luminance) rather than the R, G, and B images. The transformation from R, G, and B images to a gray-level image is defined by National Television System Committee (NTSC) and can be expressed as equation 3-1.

$$Gray = 0.299 \times \mathrm{Re}d + 0.587 \times Green + 0.114 \times Blue \qquad (3\text{-}1)$$

The local motion detection relies on the motion estimation of each block which consists of 128x128 pixels. We have to define mean-binary and noise-binary images in each block. The procedure to find the mean and noise binary images is defined as follows：

1. Calculate the mean value of grayscale block images.
2. Create a mean-binary image with 0’s where the pixels are less than or equal to the mean value and 1’s where the pixels are greater.
3. Create a noise-binary image with 0’s where the pixels are within ± 64 of the mean value and 1’s where the pixels are outside this range. Only if the noise-binary and its mean binary are operated by the AND logic, the noise existing between the mean value ± 64 is eliminated.

The procedure of block motion detection is shown in **Fig. 3-11**. Initially, we input two images with the same block and found mean-binary and noise-binary of each feature. M_A and M_B are two mean-binaries. N_A and N_B are two noise-binaries. Thereafter, M_A and N_A are shifted from - range to + range along both x and y directions (M_A' and N_A'). Taking the difference of these two mean-binary images with an Exclusive-or operator (M_A' and M_B) and putting the result (M_R) to follow two AND operators (M_R, N_B and N_A') can obtain a final binary image. Then, we compute the sum of all 1 bits in each shift step. The minimum sum presents the shift value of this block. If the shifted value of x or y is saturated to be equal to range, the range will double and shifted step will be repeated.

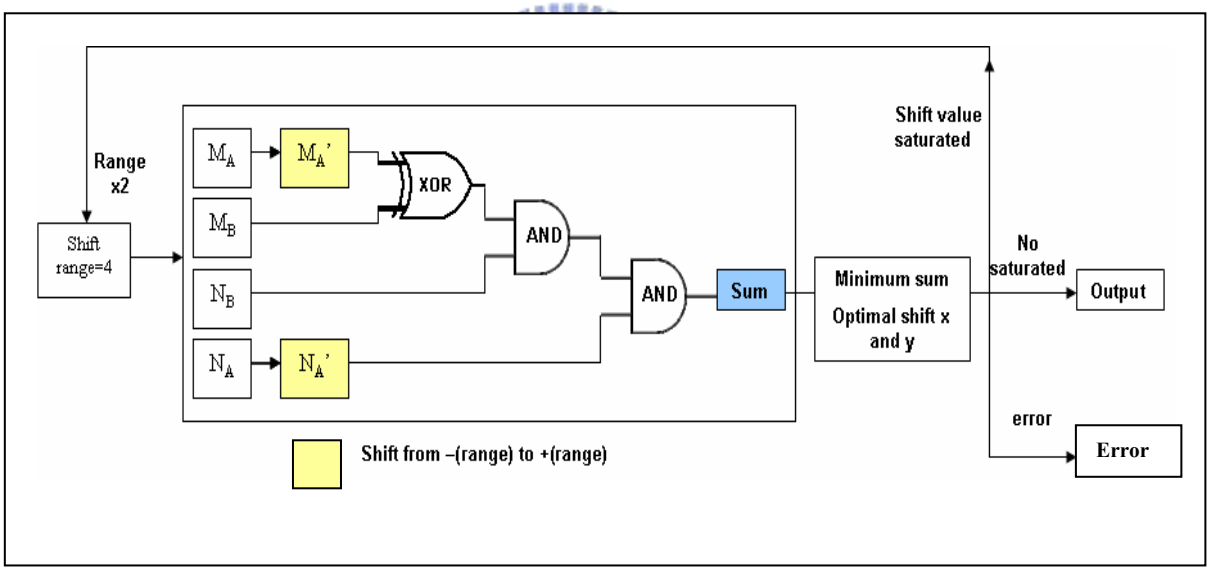


Fig. 3-11 The procedure of block motion detection.

If blocks are located in the regions with uniform color such as sky and lake or complicated objects such as trees, the minimum sum of each shifted step is zero or repeated respectively. These blocks are error. **Fig. 3-12** is the specific scenery including both uniform color and complicated objects.

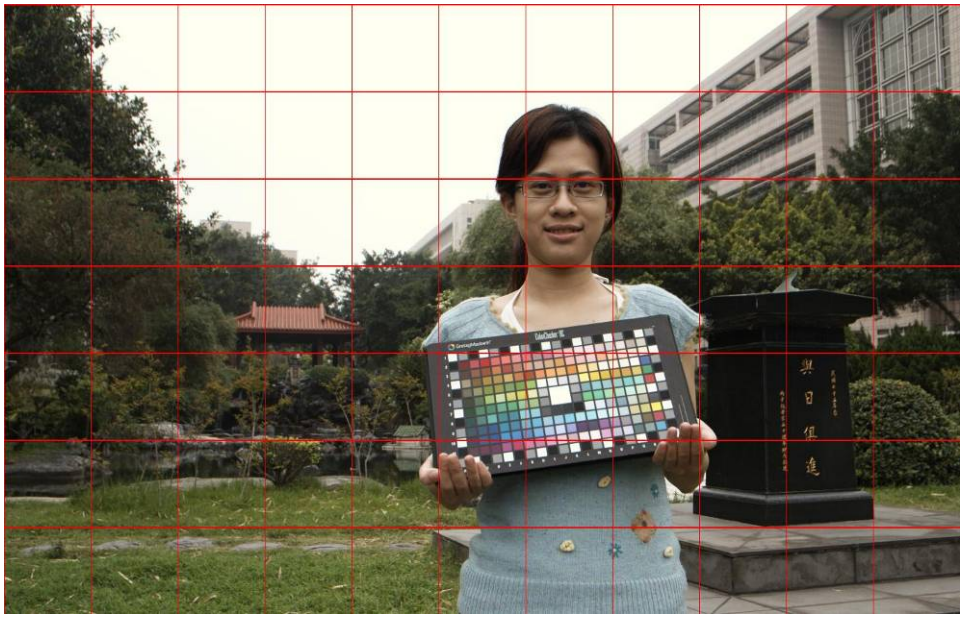


Fig. 3-12 The image is to cut by macro blocks. Some blocks are located in uniform regions (sky and lake) and some are located in complicated areas (trees).

After all the local motion vectors are completed, the error blocks are recalculated. Two methods are used to correct error motion values. The error block with repeated minimum value is replaced by the mean value averaged by adjacent blocks. The error block belongs to large uniform color regions such as sky or sea and is substituted by a method which averages the values within the standard deviation of all estimated blocks. The standard deviation is defined in equation 3-2, where μ and N are the mean of all motion values and the number of blocks except for the error blocks respectively.

$$\sigma = \sqrt{\frac{(x-\mu)^2}{N}} \qquad (3\text{-}2)$$

3.4 Summary

We proposed a fast and simple method for aligning multiple images degraded by camera shaking. The modified algorithm was improved from Greg Ward's method to achieve higher performance, less computation, less memory requirement, and could be applicable to fuse two different exposed images for HDR.