

# Local Image Registration: An Adaptive Filtering Framework

Gulcin Caner<sup>a</sup>, A. Murat Tekalp<sup>a,b</sup>, Gaurav Sharma<sup>a</sup> and Wendi Heinzelman<sup>a</sup>

<sup>a</sup>Electrical and Computer Engineering Dept. , University of Rochester, Rochester, NY 14627

<sup>b</sup>College of Engineering, Koc University, Istanbul, Turkey  
{caner, tekalp, gsharma, wheinzel}@ece.rochester.edu

## ABSTRACT

We present a novel local image registration method based on adaptive filtering techniques. The proposed method utilizes an adaptive filter to track smoothly, locally varying changes in the motion field between the images. Image pixels are traversed following a scanning order established by Hilbert curves to preserve the contiguity in the 2-D image plane. We have performed experiments using both simulated images and real images captured by a digital camera. The proposed adaptive filtering framework has been shown by experimental results to give superior performance compared to global 2-D parametric registration and Lucas-Kanade optical flow technique when the image motion consists of mostly translational motion. The simulation experiments show that the proposed image registration technique can also handle small amounts of rotation, scale and perspectivity in the motion field.

**Keywords:** Local image registration, adaptive filtering, hilbert curves

## 1. INTRODUCTION

Image registration is a fundamental step required for more complex tasks in both image processing and computer vision. Due its wide range of application areas, a large number of techniques have been proposed to solve this problem in different scenarios. Extensive surveys of image registration techniques can be found in Brown <sup>1</sup> and Zitova et al..<sup>2</sup>

Image registration techniques can be classified into two groups, i) methods that rely on only the image data, and do not take into account the underlying camera or scene geometry, and ii) methods that take into account the underlying assumptions about the image acquisition process (i.e., scene geometry, camera motion). Methods that fall into the first group are mostly designed for image processing applications, e.g., block-matching method used for video compression .<sup>3</sup> Predominant methods in the latter group are from computer vision area, e.g., plane+parallax <sup>4</sup> method for 3-D scene structure computation.

In this paper, we propose a new local image registration technique, in the first class, based on adaptive filtering techniques. Adaptive filters have been utilized successfully for system identification purposes in 1-D .<sup>5</sup> This motivates us to formulate the image registration problem as a system identification problem and build an adaptive filtering framework to solve it. Adaptive filtering has been proved to track smoothly varying changes in the system response. In order to simulate same kind of effect in the image registration problem, image pixels are traversed following a scanning order established by Hilbert Curves to preserve contiguity in the 2-D image plane and hence track a smoothly varying motion field. In the proposed adaptive filtering framework, a number of conditions are required in the filter, in order to guarantee that it models the true motion field.

The proposed image registration technique is also computationally simpler than other methods for local image registration such as the pyramid-based image registration techniques and uses only local information in the images. Both these features make the proposed technique quite applicable in practical applications (e.g., imaging sensor networks) where the computational and memory resources are scarce.

---

Send correspondence to G. Caner, email:caner@ece.rochester.edu

This work is partly supported by the National Science Foundation under grant number ECS-0428157.

## 2. THEORY OF THE ADAPTIVE FILTERING FRAMEWORK

The image registration problem can be regarded as a system identification problem by the following scenario: Pixel values in one image,  $I_1(x, y)$  can be expressed in terms of the pixel values in the other image  $I_2(x, y)$ , assuming that these two images contain overlapping views of the same scene (e.g, multiple camera views, or successive frames of a video). We can express the relation between  $I_1(x, y)$  and  $I_2(x, y)$  as a spatially varying system response,  $h_o(x, y; x_o, y_o)$ :

$$I_2(x_o, y_o) = \sum_{x, y} h_o(x, y; x_o, y_o) I_1(x, y) + e(x_o, y_o) \quad (1)$$

where  $e(x_o, y_o)$  denotes the deviations from the system response.

Using this relation, we formulate image registration problem as a system identification problem, where the system response,  $h_o(\cdot)$  is estimated in an adaptive filtering framework. Though computation of system response,  $h_o(\cdot)$  provides prediction of one image from the other, it does not guarantee the estimation of the true motion field between the images. In the next subsections, we discuss the conditions that the adaptive filter should satisfy in order to utilize the proposed adaptive filtering framework for image registration.

### 2.1. Review of 1-D Adaptive Filtering

Before explaining how 2-D adaptive filtering is utilized for image registration, we review the 1-D adaptive filtering. 1-D adaptive filtering is typically a two step process as shown in Fig. 1: i) a filtering process, where the filter coefficients,  $\hat{h}(t, t_o)$  are convolved with the input signal,  $v_1(t)$ , to produce an estimate of the desired response,  $v_2(t_o)$ , and ii) an adaptive process where the set of filter coefficients are adjusted using the resulting estimation error,  $e(t_o)$ . For the commonly used least-mean-square (LMS)<sup>5</sup> adaptation algorithm, the adaptive filtering process is given as:

$$\hat{v}_2(t_o) = \sum_{(t \in U)} \hat{h}(t, t_o) v_1(t) \quad (2)$$

$$e(t_o) = v_2(t_o) - \hat{v}_2(t_o) \quad (3)$$

$$\hat{h}(t+1, t_o) = \hat{h}(t, t_o) + \beta e(t_o) v_1(t), \forall t \in U \quad (4)$$

where  $\beta$  and  $U$  denote the adaptation step-size and the support of the 1-D filter, respectively.

Consider the scenario where the desired response  $v_2(t_o)$  is related to the input signal  $v_1(t)$ , through a system model  $h(t, t_o)$  as  $v_2(t_o) = \sum_t h(t, t_o) v_1(t)$ . Under appropriate conditions, the adaptive filter *coefficients*  $\hat{h}(t, t_o)$  closely approximate and track slow changes in  $h(t, t_o)$ . The adaptation step-size  $\beta$  determines the speed of convergence, tracking capability, and the closeness of the approximation.<sup>5</sup>

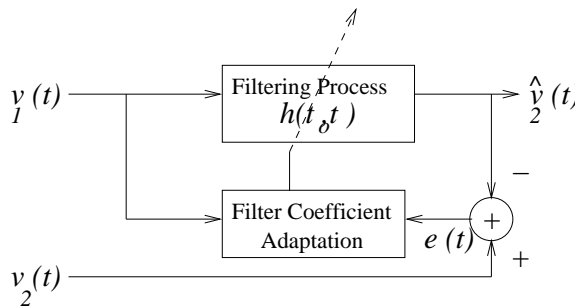


Figure 1. Least-mean-square adaptation

## 2.2. 2-D Adaptive Filtering for Image Registration

The adaptive filtering technique inherently works in 1-D domain. In order to use it for image registration (i.e. estimate  $h_o(x, y; x_o, y_o)$ ), it should be extended properly to 2-D domain. For two-dimensional images, the adaptive filter takes the format of a 2-D finite impulse response (FIR) filter,  $h(x, y)$ . Figure 2 shows the support of the filter,  $R$ , on the reference image,  $I_1(x, y)$ . Using the 2-D notations, the 2-D LMS adaptation algorithm can be written as:

1) Filter output (Prediction phase):

$$\hat{I}_2(x_o, y_o) = \sum_{(x, y) \in R} \hat{h}_b(x, y; x_o, y_o) I_1(x, y) \quad (5)$$

2) Estimation error:

$$e(x_o, y_o) = I_2(x_o, y_o) - \hat{I}_2(x_o, y_o) \quad (6)$$

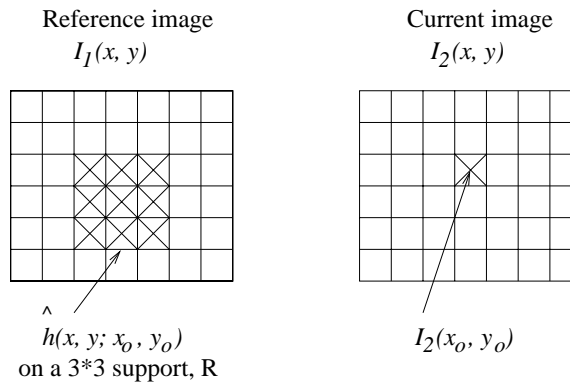
3) Filter adaptation (Update phase):

$$\hat{h}_a(x, y; x_o, y_o) = \hat{h}_b(x, y; x_o, y_o) + \beta e(x_o, y_o) I_1(x, y), \forall (x, y) \in R \quad (7)$$

4) Initializing the filter for the next pixel,  $(x_n, y_n)$ :

$$\hat{h}_b(x, y; x_n, y_n) = \hat{h}_a(x, y; x_o, y_o) \quad (8)$$

where  $(x, y) \in R$  and  $\beta$  is the adaptation step-size. The subscripts 'b' and 'a' denote 'before' and 'after' adaptation, respectively. Under the appropriate conditions, (i.e., the right step-size and the right filter size)  $\hat{h}(x, y; x_o, y_o)$  converges to the system model,  $h_o(x, y; x_o, y_o)$  which maps the reference image to the current image.

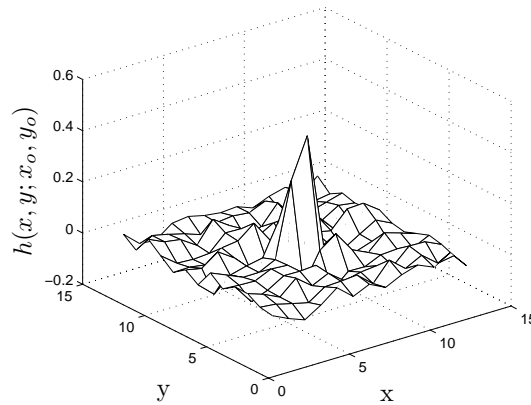


**Figure 2.** 2-D adaptive filtering for images

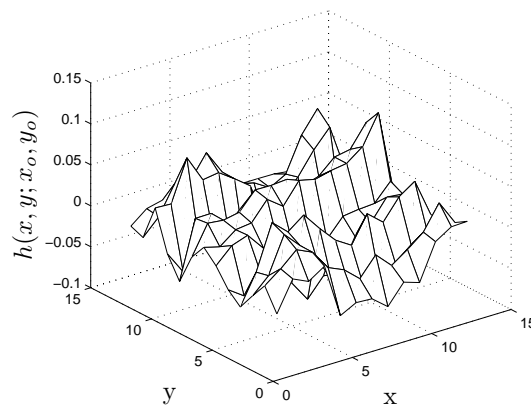
## 2.3. Motion Modeling Through Adaptive Filter

Although the estimation of system response,  $h_o(\cdot)$  can provide a prediction of  $I_2(x, y)$  from  $I_1(x, y)$ , it models the true motion field between  $I_1(x, y)$  and  $I_2(x, y)$  only if the following conditions are satisfied: i) the system response should have a single peak, and ii) its elements should sum to 1.0.

Figures 3 and 4 show two system responses, estimated for two different image pairs, respectively for the image pair shown in Figure 5 and for the image pair shown in Figure 6. As observed in Figure 3, the system response has a single peak and hence models the true motion field between the images shown in Figure 5. The system response shown in Figure 4 has more than one peak, because the input images are two different images. Figure 7 shows the absolute registration error image, after the proposed 2-D adaptive filtering method is applied to these uncorrelated images. From Figures 4 and 7, we conclude that the system response,  $h_o(\cdot)$  does not model the true motion field between the uncorrelated images even though it can be used to predict one image from another.



**Figure 3.** 2-D adaptive filter estimated for flower images



**Figure 4.** 2-D adaptive filter estimated for two uncorrelated images

### 3. IMPLEMENTATION

In this section, we discuss the issues relating to the implementation of the proposed adaptive filtering framework for image registration. In 1-D adaptive filtering, time provides a natural order for 1-D signals. However, for 2-D images, a scanning order should be established in order to apply the proposed 2-D adaptive filtering technique. This order can be established through the use of space-filling curves that map multi-dimensional data to 1-D space. In the proposed adaptive filtering framework, in order for the adaptive filter to converge and track spatially varying changes in the motion field, the desirable space-filling curve should have the following properties: i) adjacent pixels in 1-D space should be adjacent in 2-D image plane, and ii) consecutive pixels in 1-D space should be within the same neighborhood in 2-D image plane. Hilbert Curves<sup>6</sup> provide these conditions, and therefore used to map the 2-D image plane to 1-D. Figure 8 shows a Hilbert curve for a 2-D region of size  $16 \times 16$ .

In the standard 2-D LMS adaptation algorithm, one pixel information is utilized to estimate the adaptive filter coefficients at the corresponding location in the current image. In other terms, with one equation, a set of filter coefficients are computed (i.e., more unknowns than number of equations). This results in a number of valid solutions, known as “aperture problem” in the optical flow estimations.<sup>3</sup> To regularize the problem and avoid indiscriminate changes in the filter coefficients, we utilize block LMS adaptation algorithm<sup>5</sup> rather



**Figure 5.** Flower images



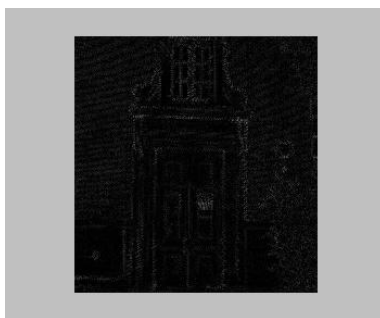
**Figure 6.** Two uncorrelated images

than standard LMS. In this technique, a uniform motion constraint is applied over a block of pixels around the current pixel, for which the adaptive filter is estimated.

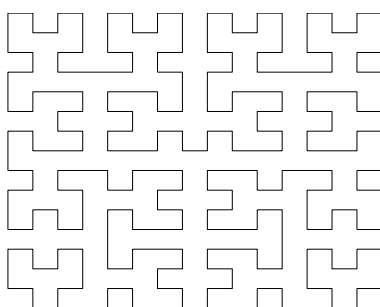
Large displacements among the images may require a very large adaptive filter. However, filter size cannot be simply increased in the adaptation process because it will affect the convergence characteristic of the filter and increase the computational burden. In order to be able to use a filter with a constant size in the adaptation process, and still keep tracking large locally varying changes in the mis-registration, adaptive filter is shifted to its center of mass after each update, and integer pixel shifts are saved along with the filter coefficients.

Another important variable in the proposed adaptive filtering framework is the adaptation step-size,  $\beta$ . This variable determines the rate at which the adaptive filter coefficients  $h(x, y; x_o, y_o)$  are adapted. However, there is a trade-off between the convergence characteristic and tracking capability of the filter. A large  $\beta$  can provide a better tracking of the system response, when there is a significant variation in the system model,  $h_o(x, y; x_o, y_o)$ , though it may result in large gradient estimation noise,<sup>5</sup> and prevent the filter to converge. In order to provide fast tracking capability and also prevent gradient noise amplification, we apply normalized version of the LMS adaptation algorithm.<sup>5</sup>

The proposed image registration technique is designed to solve local mis-registration among the images. Therefore, it should follow an initial global alignment phase in order to handle large displacements between the images and coarsely align them. In the experiments, initial global alignment was performed following two approaches: a) 2-D parametric registration,<sup>7</sup> and b) fitting a parametric motion model into a number of correspondent image features in the images.



**Figure 7.** Registration error after the proposed registration method



**Figure 8.** Hilbert Curve on a  $16 \times 16$  square

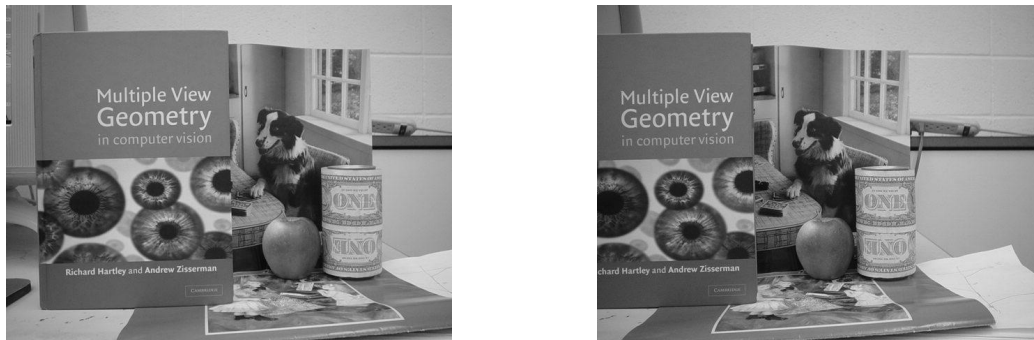
#### 4. EXPERIMENTAL RESULTS

We applied the proposed image registration technique on both real and simulated images. Figure 9 shows two input images, which are captured by a moving digital camera. The scene consists of multiple planes, thus invalidating the planar scene assumption. The spatially-locally varying mis-registration among the two images results from camera motion, and differences in lighting due to deviations from lambertian assumptions.

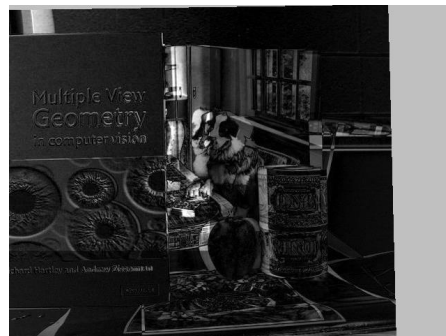
Before applying the proposed local image registration method, the input images are aligned globally with respect to the book on the left side of the 3-D scene. This is achieved by selecting a number of correspondent feature points on the book object manually, and fitting an eight parametric motion model into the correspondent points. After warping one input image towards the other with the estimated motion parameters, the absolute registration error image, shown in Figure 10 is computed. The estimated motion parameters are then used to update the adaptive filter for the first pixel on the scan order of the current input image. The adaptive filter is then updated following the scan order established by Hilbert Curves, using the proposed 2-D adaptive filtering technique.

The absolute registration error image, shown in Figure 11 is computed by using the following set of parameters in the adaptive filtering framework: 1) adaptation step size is set to 0.2, 2) a constant size adaptive filter ( $13 \times 13$ ) is used in the adaptation process 3) block size for the block LMS is set to  $(3 \times 3)$ , 4) number of iterations per image is set to 5. Both of the Figures 10 and 11 are contrast-enhanced (i.e., multiplied by 2) for better presentation. The pSNR values of the error images after the initial global alignment and the proposed registration method are 17.93 dB and 31.24 dB, respectively. The proposed technique is also compared with hierarchical Lucas-Kanade technique,<sup>8</sup> using a window size of  $(9 \times 9)$  that results in the highest pSNR. The corresponding pSNR of the absolute registration error image is computed to be 23.35 dB (i.e., number of pyramid levels is 3).

In order to confirm that the estimated adaptive filter models the true motion, we check the filter coefficients, (i.e., they should have a single peak and they should sum to 1.0). Figure 12 shows the converged filter at an arbitrary location. Figure 13 shows a 2D map of integer pixel shifts. The camera motion resulting in the image motion between the two images can be observed from this figure. The pixel locations, where the integer pixel shifts are not uniform, indicate the occluded regions, where the motion field cannot be estimated. There are some overlaps observed in Figure 13. This is because computation of Hilbert curve requires the 2-D region to be a square. In the implementation, input images are divided into overlapping squares, and a Hilbert Curve is computed for each square. Then, the proposed 2-D adaptive filtering technique is performed on each square iteratively.



**Figure 9.** left:Reference input image, right:Current input image for multiple planar object

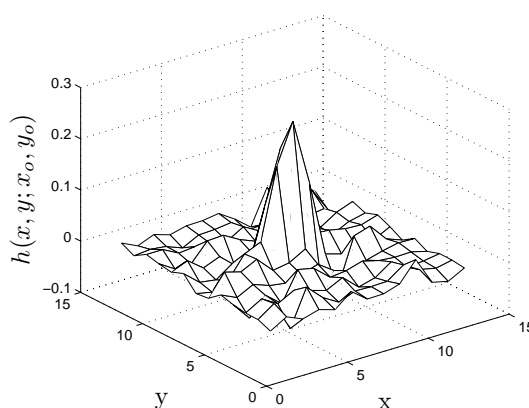


**Figure 10.** Registration error after global alignment

The proposed adaptive filtering method is also applied to simulated images, in order to evaluate its performance when the input images have large rotation/scale differences. Figure 14 shows two such images, where the left input image is built from the right input image. Figure 15 shows the absolute difference image between the images with a pSNR of 17.96 dB. The proposed 2-D adaptive filtering technique for image registration is applied to the images, with the same parameters used in the first experiment. Figure 16 shows the absolute registration error image with a pSNR of 31.50 dB. Both Figures 15 and 16 are contrast-enhanced (i.e., multiplied by 2) for better presentation.



**Figure 11.** Registration error after the proposed registration method



**Figure 12.** 2-D adaptive filter for multiple planar object

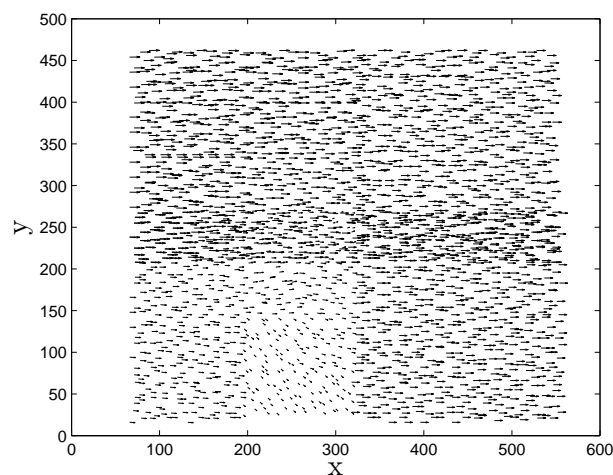
## 5. CONCLUSION

In this paper, we presented a new local image registration technique using 2-D adaptive filtering techniques. We set some conditions on the adaptive filter, (i.e., it should have a single peak) in order to estimate the true motion field. Using Hilbert Curves to traverse the input images, we guaranteed a slowly-varying system response and achieved filter convergence together with improved tracking capability. We performed extensive experiments using both real images and simulated images. Experiment results indicate that the proposed technique can register images with locally varying motion even when the input images have small amount of rotation/scale/perspectivity differences. The proposed registration method also provides higher pSNR's when compared to 2-D parametric registration and Lucas-Kanade local image registration technique.

## REFERENCES

1. L. G. Brown, "A survey of image registration techniques," *ACM Computing Surveys* **24**, pp. 325–376, Dec. 1992.
2. B. Zitova and J. Flusser, "Image registration methods: A survey," *Image and Vision Computing* **21**(11), pp. 977–1000, 2003.
3. A. M. Tekalp, ed., *Digital Video Processing*, Prentice Hall, Upper Saddle River, NJ, 1995.





**Figure 13.** 2D map of integer pixel shifts

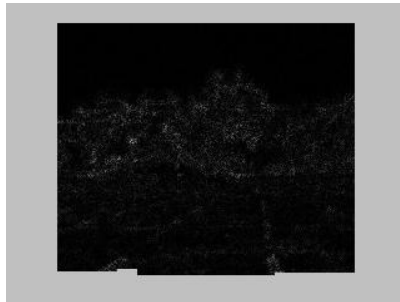


**Figure 14.** left:Reference input image, right:Current input image for simulation experiment

4. H. Sawhney, "Simplifying motion and structure analysis using planar parallax and image warping," *International Conference on Pattern Recognition* , pp. A403–A408, 1994.
5. S. Haykin, *Adaptive Filter Theory*, Prentice Hall, NJ, fourth ed., 2002.
6. H. Sagan, *Space-filling curves*, Springer, Berlin, 1994.
7. J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani, "Hierarchical model-based motion estimation," *European Conference on Computer Vision* , pp. 237–252, May 1992.
8. B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," *Proc. DARPA Image Understanding Workshop* , 1981.



**Figure 15.** Simulation experiment: Difference between the input images



**Figure 16.** Simulation experiment: Registration error after the proposed registration method