Detecting Changes in 3-D Shape using Self-Consistency

(to appear in the Proceedings of CVPR 2000)

Yvan G. Leclerc, Q.-Tuan Luong, Pascal V. Fua, Koji Miyajima
SRI International, Menlo Park, CA
EPFL, Lausanne, Switzerland
Laboratory for Information Technology, NTT DATA Corporation
<leclerc,luong@ai.sri.com, fua@lig.di.epfl.ch, miya@rd.nttdata.co.jp>

Abstract[1]

A method for reliably detecting change in the 3-D shape of objects that are well-modeled as single-value functions $z = f(x,y)$ is presented. It uses an estimate of the accuracy of the 3-D models derived from a set of images taken simultaneously. This accuracy estimate is used to distinguish between significant and insignificant changes in 3-D models derived from different image sets. The accuracy of the 3-D model is estimated using a general methodology, called self-consistency, for estimating the accuracy of computer vision algorithms, which does not require prior establishment of "ground truth". A novel image-matching measure based on Minimum Description Length (MDL) theory allows us to estimate the accuracy of individual elements of the 3-D model. Experiments to demonstrate the utility of the procedure are presented.

1 Introduction

Detecting where an object's 3-D shape has changed requires not only the ability to model shape from images taken at different times, but also the ability to distinguish significant from insignificant differences in the models derived from two image sets.

In this paper, we present an approach to distinguishing significant from insignificant changes based on a novel methodology called *self-consistency* [1]. This methodology allows us to estimate, for a given 3-D reconstruction algorithm and class of scenes, the expected variation in the 3-D reconstruction of objects as a function of viewing geometry and local image-matching quality (referred to as a “score” below).

Differences between two 3-D reconstructions of an object that exceed this expected variation for a given significance level are deemed to be due to a change in the object’s shape, while those below this are deemed to be due to uncertainty in the reconstructions.

Although our methodology for change detection is quite general, the experiments we conducted are based on two simplifying assumptions.

First, we use a specific class of objects: terrain (both rural and urban) viewed from above using aerial imagery. We take advantage of the special nature of terrain to simplify the problem. The 3-D shape is modeled as a single-valued function $z = f(x,y)$, where z represents elevation above the ground plane (x,y). Second, we assume that all the camera parameters for all the images are known in a common coordinate system, which we obtain by bundle adjustment over all the images. Together, these two assumptions reduce the problem of detecting changes in 3-D shape to that of finding point-by-point significant differences in scalar values.

In the remainder of this paper we present a brief overview of previous work in change detection, the self-consistency methodology, and its application to detecting changes. We then present results on rural and urban scenes to demonstrate the utility of our approach

2 Previous work in change detection

Change detection is an important task in computer vision that has been addressed early at the image intensity level [2, 3] where it is still a topic of interest [4] with many other papers in between). However, comparing intensity values is not very effective because such changes don't necessarily reflect actual changes in shape, but could be caused by changes in viewing and illumination conditions or even in reflectance (e.g., seasonal changes). Although it has been attempted, this is not easy to take them into account at this level. For man-made objects such as buildings, higher-level comparisons have been proposed, based on feature organization [5], and 3-D model [6-8]. These

[1] This work was sponsored in part by the Defense Advanced Research Projects Agency under contract F33615-97-C-1023 monitored by Wright Laboratory. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency, the United States Government, or SRI International.

specialized approaches are the most successful, but are not applicable to more general objects like natural terrain.

A few of the ideas needed for general change detection in shape are found in other areas of computer vision. In work on tracking (see for instance [9] which deals with small bodies in a natural environment), statistics have been computed during a learning phase and then used to differentiate between significant and insignificant changes. Of course, the problem is simplified by the fact that the camera is stationary, whereas we want to deal with various viewpoints. In mobile robotics (see for instance [10]), the problem of fusing several general 3-D maps to maintain representations of the environment of a robot has been cast in a statistically rigorous framework taking into account uncertainties,. However, the specific issue of change detection has not been addressed there.

3 The Self-Consistency Methodology for change detection.

3.1 Self-consistency of stereo and uncertainty

In order to distinguish significant from insignificant changes, it is necessary to have some kind of measure of the uncertainty of the algorithm output.

The self-consistency methodology [1, 12] makes it possible to measure the expected variation in the output of a computer vision algorithm as a function of viewing geometry and contextual measures, for a given algorithm and a given class of scenes. This expected variation can be expressed as a probability distribution that we call the self-consistency distribution. It is computed from sets of images by independent applications of the algorithm to subsets of images, and in particular does not require the knowledge of ground truth.

A stereo algorithm attempts to find *matches* consisting of a pair of points, one in each image, that both correspond to the same surface element in the world. Though there are many variations on this theme, a typical stereo algorithm starts with a point in the first image. It searches for a matching point in the second image that maximizes some measure of the similarity of the image in the neighborhood of the two points (the *score*).

When the internal and external camera parameters (the full projection matrix for a pinhole camera) of a pair of images are known, given a match and the camera parameters, we can estimate the 3-D coordinate of the surface element by triangulation.

Now, imagine having several images of a static scene, all with known camera parameters. If we apply the same stereo algorithm to all pairs of images of this scene we can expect to get reconstructions that are quite similar in some places (where there is good texture and the surface is locally planar, for example) but quite different in others. The self-consistency distribution in this case is the distribution of differences in the triangulated 3-D coordinates for matches that belong to the same surface element, for all pairs of images of a given static scene. The key step in estimating this self-consistency distribution is identifying those matches from different image pairs for which the stereo algorithm is asserting belong to the same surface element.

In our application to terrain, we can take advantage of the special form of the surface we are estimating, $z = f(x,y)$, to find matches that necessarily correspond to the same surface element. Namely, matches that have the same (x,y) coordinates for their 3-D triangulation correspond to the same surface element. The histogram of the differences between the z coordinates for such matches (appropriately normalized) is what we call the *common-xy-coordinate self-consistency distribution*. This distribution is also particularly appropriate for object-centered surface-reconstruction algorithms, such as our deformable meshes [13, 14].

The normalization mentioned above, described in more detail in [1, 12], divides the measured difference by its expected variance for the given camera parameters and an assumed variance of 1 pixel in match coordinates. Consequently, a normalized difference of 1 unit corresponds to a difference in disparity of about 1 pixel for that particular pair of images.

3.2 Summarizing self-consistency with the MDL score

We would like to be able to compare a reconstruction of a scene created with as few as two images of the scene taken at time t_1 against a reconstruction of the same scene created with as few as two images taken at a different time, t_2. There is not enough data in these conditions to compute the self-consistency distribution.

We address this difficulty by (a) computing self-consistency distributions as a function of a class of scenes and a particular algorithm, (b) computing self-consistency as a function of an appropriate score.

For a sufficiently restrained class of scenes (such as the set of images of the rural scenes or the set of images of the urban area shown in section 4), the self-consistency distribution remains reasonably constant over many scenes taken at the same instant. We can then use the average of these distributions to represent the self-consistency distribution of new images of a new scene within the same class.

Furthermore, we can use this average distribution to predict the expected variation in reconstruction when we have only a single image pair of a new scene within the same class. The idea is to use the score as a predictor of self-consistency. Since self-consistency is correlated with the quality of reconstruction, we would hope that with a suitable score, the reconstructions will be similar for places where the match score was good and dissimilar otherwise. The use of a score which has this property is essential for the proposed change detection method to work.

In this paper we use a score based on Minimum Description Length (MDL) theory developed by us. It has a stronger correlation with self-consistency than other scores we have examined [1], in particular the SSD residual. The score is described in detail in Appendix A.

3.3 The change detection algorithm

In the first stage, we run the stereo algorithm on a large number of subsets of images of the same class as those in which we want to perform change detection. We use a bucketing method to find all the common-xy-coordinate matches (pairs of matches for which the 3-D reconstruction has the same (x,y) value within a threshold). Each such pair is accumulated in a scatter diagram (see Figure 1(a)) in which the x-coordinate of is the larger of the scores for the two matches, and the y-coordinate is the normalized difference between their triangulated z coordinates.

We then extract the significance level curves for the values of significance s%(or in other words, percent confidence in the significance of the change) which we plan to use. For a given value of the score (the x-axis), this is the normalized difference below which s% of the common-xy-coordinate match pairs with that score lie.

In the second stage, we use the pre-computed significance level curves to judge whether a pair of matches derived from images taken at different times is significantly different. We find, using the same technique as before, the common-xy-coordinate matches where each match originates from a different instant, compute the larger of their scores and the normalized difference between triangulated z coordinates.

If, for that score, this distance is above the significance level s% then the pair of matches is deemed to be a difference significant with confidence s%.

4 Experimental Results

4.1 Self-consistency Distributions

In Figure 1 we see several representations of the common-xy-coordinate self-consistency distribution. The distribution was derived from the application of the point-by-point stereo algorithm [11] to 17 rural scenes, each consisting of 5 aerial images, for a total of 17*10=170 image pairs. The images had a ground resolution of approximately 15cm.

The bar graph of Figure 1(b) is the histogram of the normalized difference in the z coordinate of the triangulation of all common-xy-coordinate match pairs. One can see from this histogram that the mode of the differences is about 1 normalized unit. The curve is the integral of this graph, or the cumulative distribution function. One can see from this that about 90% of the match pairs have normalized z differences below 2 units.

Each point in Figure 1(a) corresponds to a common-xy-coordinate match pair. The x-coordinate of the point in the diagram is the larger of the scores for the two matches, and the y-coordinate is the normalized difference between their triangulated z coordinates. The curve in Figure 1(a) is the 99% significance level. Note that significance level increases as the score increases, indicating that matches with larger scores are less self-consistent than matches with a lower score, an indication of the quality of our MDL score. The drop that we observe for positive values of the score is due to the fact that there are only few common-xy-coordinate match pairs with positive scores, so that calculations done with those values are not statistically meaningful.

In Figure 2 we see the common-xy-coordinate self-consistency distribution for our deformable mesh algorithm applied to the same images. Note that it is significantly more self-consistent than the distribution for the stereo algorithm. This is as expected, since the deformable mesh algorithm was specifically designed to provide highly accurate reconstructions of terrain.

In Figure 4 we see the common-xy-coordinate self-consistency distribution for the stereo algorithm, with a correlation-window size of 7x7 pixels, applied to 6 urban scenes, each consisting of 4 images (a total of 6*6=36 image pairs) at a ground resolution of approximately 50cm. Note that the cumulative distribution indicates that about 90% of the match pairs have a normalized difference below 2 units.

Figure 5 shows the distribution for the same algorithm, but using 15x15 image windows. Note that the cumulative distribution indicates that a much higher percentage, about 98%, of the match pair have a normalized difference below 2 units.

Note also that the significance level is, overall, higher than for the rural scenes. This is because there are many repeating structures in rural scenes. Thus, a score based purely on the similarity of image windows cannot always distinguish between good and bad matches. However, we see that larger window sizes allow the score to distinguish good from bad matches more often. We are currently exploring the use of a two-dimensional score vector, in which the second element is the score of the second-best match along an epipolar line.

4.2 Change detection results

In Figure 3 we show the changes detected in one of the rural scenes mentioned above, using the deformable mesh algorithm. In Figure 3(a) we see one of 5 images of the scene taken in 1995. The dark diagonal near the center of the image is a dried creek bed. In Figure 3(b) we see one of 5 images of the same area taken in 1998. The dried creek bed has been filled in with dirt, creating a change in elevation of about 1 meter. We applied the deformable mesh algorithm to one pair of images taken in 1995. We then compared this to the deformable mesh derived from one pair of images taken in 1998. Vertices that were deemed to be significantly different (above the 99% level of the self-consistency distribution of Figure 2(a)), are overlaid as white cross on the image in Figure 3(c), which is a magnified view of the dried creek bed of Figure 3(a).

We have also applied our algorithm to forested areas of the same rural scene. Although the normalized differences in z-coordinates is sometimes much larger (10 meters), no changes were deemed significant. Indeed, it is known that the mesh algorithm performs poorly on images of tree canopies, so that reconstruction noise could account for the differences.

In Figure 6 we show the changes (significant differences in z) detected in one of the urban scenes mentioned above, but this time using the stereo algorithm with15x15 windows. In Figure 6(a) we see one of 4 images taken at time 1. Note the new building near the center of the image. In Figure 6(b) we see one of the images taken at time 2. In Figure 6(c) we see the significant differences between the matches derived from a single pair of images taken at time 1 and the matches derived from a single pair of images taken at time 2, for a significance level of 99.99%. In Figure 6(d) we have merged the significant differences between each pair of images at time 1 and each pair of images at time 2. Note that virtually all differences are at the location of the new building.

For comparison, we show what would happen if we simply thresholded the normalized difference in triangulated z coordinates. In Figure 7(a) we show the differences between a single pair of images at time 1 and a single pair of images at time 2, for a threshold of 3 units. In Figure 7(b) we show the differences for a threshold of 6 units, which is the average difference detected in Figure 4. This value of the threshold is the highest one for which no correct changes are missed, yet it is seen that many incorrect changes are still detected. In Figure 7(c) we see the union of the differences for all image pairs.

In Figures 8 and 9 we see the results of change detection for two other urban scenes, one with a new building, the other without significant changes.

5 Summary and conclusions

We have extended the self-consistency methodology to deal with varying scenes, resulting in a reliable and robust method for detection of changes in 3-D shape. The components include: a new image-matching measure called the coding loss; a novel framework for estimating the accuracy and reliability of shape modeling procedures, applicable to other stereo reconstruction procedures; a method for normalizing the effects of camera parameters and their co-variances; and a procedure for applying the self-consistency framework. This framework could be used with other 3-D attributes.

The experimental results based on the above framework are promising. We applied our framework to reliably detect quite small changes in terrain (some corresponding to less than a pixel in disparity) in rural scenes using a deformable mesh algorithm, and large changes in urban scenes, which are quite difficult because of occlusions, using a traditional stereo algorithm.

Appendix A: The MDL-Based Score

The problem with the traditional sum-of-squared-differences (SSD) score is that it's ambiguous. That is, a low SSD score can occur not only when the facet is correctly located (as expected), but also when the facet is incorrectly located and the terrain is spatially uniform.

Intuitively, then, we want an image-matching measure that is low only when the match between the predicted and observed pixel values is close *and* the pixel values form a sufficiently complex pattern that it is unlikely to be matched elsewhere.

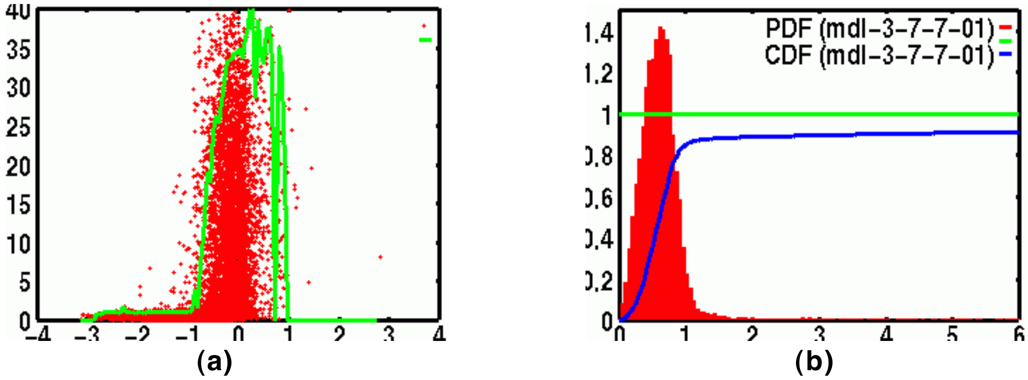


Figure 1. (**a**) Scatter diagram for 170 image pairs of rural scenes. Each point represents a pair of matches that have a common point in one image. The x-coordinate of the point is the larger of the scores of the two matches. The y-coordinate is the normalized difference in the z coordinate of the triangulations of the matches. Curve: 99% significance level, which is the largest normalized z difference for which 99% of the common-xy-coordinate matches with a given score lie below. (**b**) The histogram of the normalized z differences for all common-xy-coordinate matches. Curve: The cumulative distribution of the histogram.

We have developed a measure that satisfies this intuitive requirement , which we call the *coding loss*. It is the difference between two different ways of encoding the pixels in the correlation windows. It is based on Minimum Description Length (MDL) theory [15]. In MDL theory, quantized observations of a random process are encoded using a model of that process. This model is typically divided into two components: a parameterized predictor function $\mathrm{M}(\mathbf{z})$ and the residuals (differences) between the observations and the values predicted by that function. The residuals are typically encoded using an i.i.d. noise model [16]. MDL is basically a methodology for computing the parameters $\mathbf{z}$ that yield the optimal lossless code length for this model and for a given encoding scheme.

Given N images (2 for stereo), let M be the number of pixels in the correlation window and let g_i^j be the image gray level of the i^{th} pixel observed in image j. For image j, the number of bits required to describe these gray levels depends on the model we choose. The simplest coding model, which we use here, is to encode the pixels as IID white noise. The encoding cost in this case can be approximated by:

$$C_j = M \cdot (\log \sigma_j + c)$$

where σ_j is the measured variance of the $\{g_i^j\}_{1<i\leq N}$ and $c = \frac{1}{2}\log(2\pi e)$.

Alternatively, these gray levels can be expressed in terms of the mean gray level $\overline{g}_i$ across images and the deviations $(g_i^j - \overline{g}_i)$ from this average in each individual image. The cost of describing the means, can be approximated by

$$\overline{C} = M \cdot (\log \overline{\sigma} + c)$$

where $\overline{\sigma}$ is the measured variance of the mean gray levels. Similarly the coding length of describing deviations from the mean is given by

$$C_j^d = M \cdot (\log \sigma_j^d + c)$$

where σ_j^d is the measured variance of those deviations in image j. Note that, because we describe the mean across the images, we need only to describe $N-1$ of the C_j^d. The description of the N^{th} one is implicit.

The MDL score is the difference between these two coding lengths, normalized by the number of samples, that is

$$Score = \overline{C} + \sum_{1\leq j\leq N-1} C_j^d - \sum_{1\leq j\leq N} C_j$$

When there is a good match between images, the $\{g_i^j\}_{1<i\leq N}$ have a small variance. Consequently the C_j^d should be small, $\overline{C}$ should be approximately equal to any of the C_j and $Score$ should be negative. However, C_j can only be strongly negative if these costs are large enough, that is, if there is enough texture for a reliable match. See [17] for more details.

References

1. Leclerc Y.G., Luong, Q.-T. and Fua, P. *Measuring the self-consistency of stereo algorithms,* in ECCV 2000. Dublin, Ireland.
2. Quam, L.H., *Computer Comparison of Pictures*, in *Computer Science*. 1971.
3. Lillestrand, R.L., *Techniques for Change Detection*. IEEE TC, 1972. **21**(7): p. 654-659.
4. Rosin, P.L. *Thresholding for Change Detection*. in *ICCV*. 1998.

5. Sarkar, S. and K.L. Boyer, *Quantitative Measures of Change Based on Feature Organization: Eigenvalues and Eigenvectors*. CVIU, 1998. **71**(1): p. 110-136.

6. Bejanin, M., *et al. Model Validation for Change Detection*. in *WACV*. 1994.

7. Huertas, A. and R. Nevatia. *Detecting Changes in Aerial Views of Man-Made Structures*. in *ICCV*. 1998.

8. Mukawa, N., K. Miyajima, and S. Watanabe. *Detecting Changes of Buildings from Aerial Images Using Shadow and Shading Model*. in *ICPR*. 1998.

9. Boult, T. *Frame-rate multi-body tracking for surveillance*. in *DARPA IU Workshop*. 1998.

10. Ayache, N., *Artificial Vision for Mobile Robots*. 1991: MIT Press.

11. Fua, P., *A Parallel Stereo Algorithm that Produces Dense Depth Maps and Preserves Image Features*. MVA , 1993. **6**(1).

12. Leclerc Y.G., Luong, Q.-T. and Fua, P.*: A Novel Approach to Characterizing the Accuracy and Reliability of Point Correspondence Algorithms*. in *DARPA Image Understanding Workshop*. 1998. Monterey, CA: Morgan Kauffman.

13. Fua, P. and Y.G. Leclerc, *Object-Centered Surface Reconstruction: Combining Multi-Image Stereo and Shading*. International Journal of Computer Vision, 1995. **16**: p. 35--56.

14. Fua, P. and Y.G. Leclerc, *Taking Advantage of Image-Based and Geometry-Based Constraints to Recover 3--D Surfaces*. Computer Vision and Image Understanding, 1996. **64**(1): p. 111--127.

15. Rissanen, J., *Minimum-Description-Length Principle*. Encyclopedia of Statistical Sciences, 1987 . **5**: p. 523--527.

16. Leclerc, Y.G., *Constructing Simple Stable Descriptions for Image Partitioning*. International Journal of Computer Vision, 1989. **3**(1): p. 73-102.

17. Leclerc Y.G., Luong, Q.-T. and Fua, P. *A Framework for Detecting Changes in Terrain*. in *DARPA Image Understanding Workshop*. 1998. Monterey, CA: Morgan Kauffman.

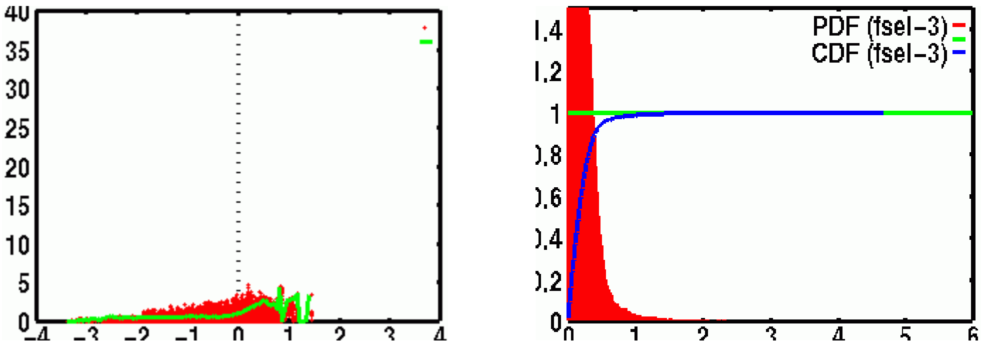
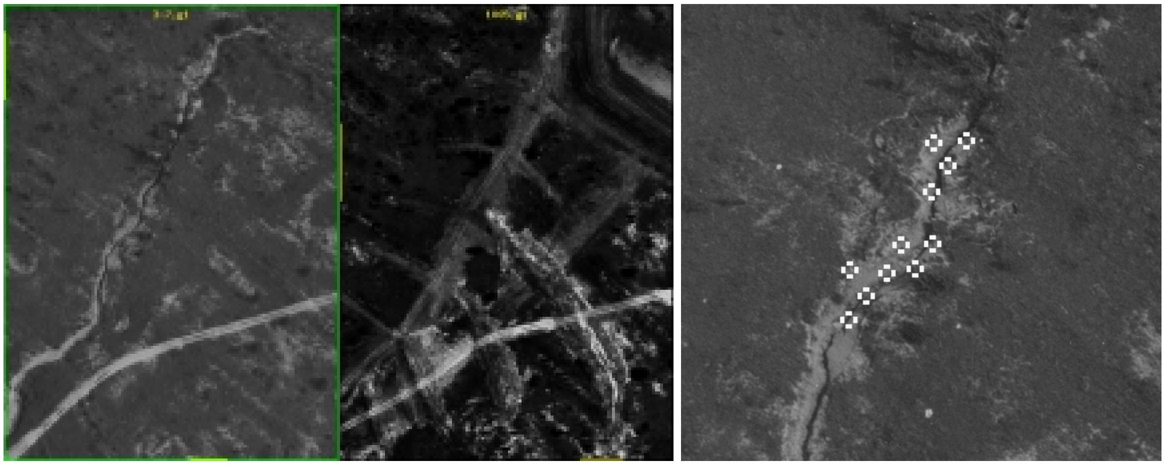


Figure 2. Scatter diagram and histograms for the same images as above, but for the deformable mesh algorithm.



(a) (b) (c)

Figure 3. Changes detected in one of 17 rural scenes. **(a)** One of 5 images of a dried creek bed taken in 1995. **(b)** One of 5 images of the same area taken in 1998. Note that the dried creek bed has been filled in with dirt, causing a change in elevation of about 1 meter. **(c)** The significant differences found between the deformable-mesh model created with one pair of 1995 image and a model created using one pair of 1998 images, using the self-consistency distribution of Figure 2.

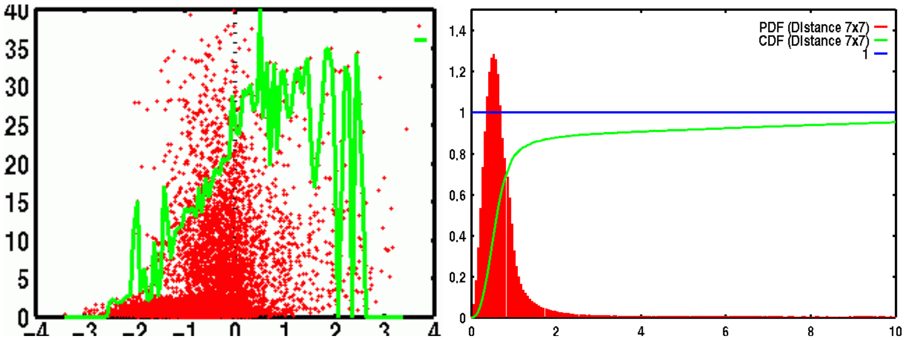


Figure 4. Scatter diagram and self-consistency distribution graphs for 7x7 windows of the 6 urban scenes, each comprising 4 aerial images.

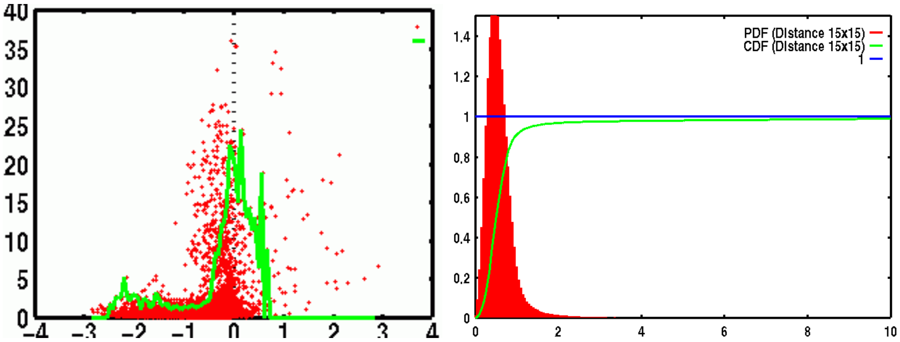
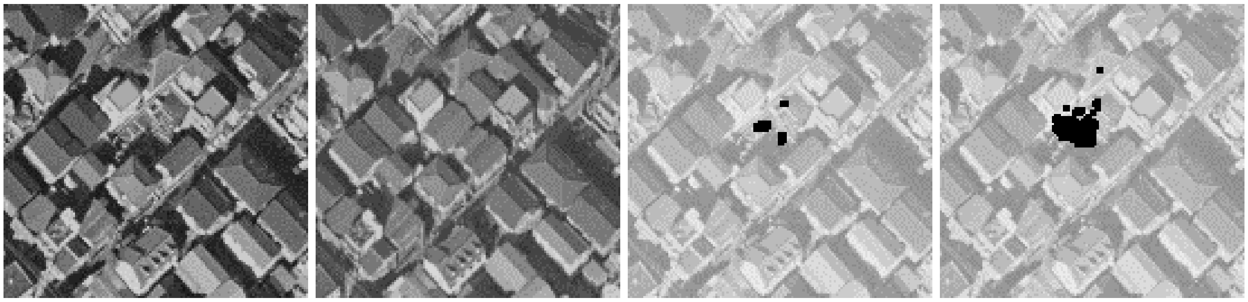
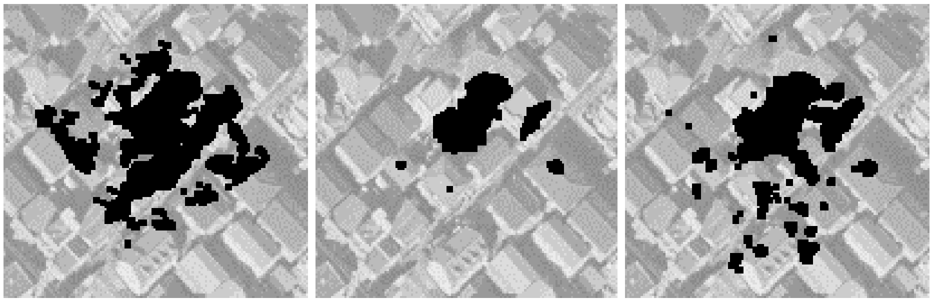


Figure 5. Scatter diagram and self-consistency distribution graphs for 15x15 windows of the same urban scenes as above.



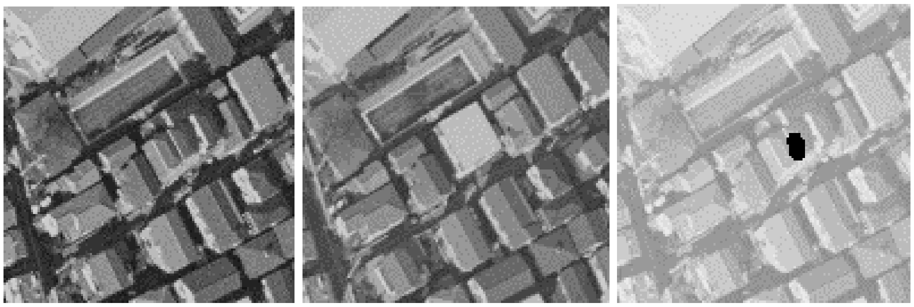
(a) **(b)** **(c)** **(d)**

Figure 6. The first urban scene. **(a)** One of 4 images taken at time 1. **(b)** One of 3 images taken at time 2. Note that there is a new building near the center of the image. **(c)** The significant differences between matches derived from one pair of images taken at time 1 and matches derived from one pair of images taken at time 2. **(d)** The union of all significant differences found between matches derived from all pairs of images taken at time 1 against all pairs of images taken at time 2.



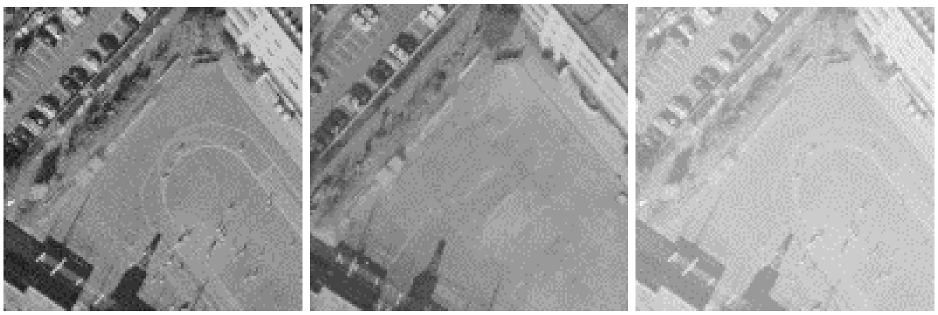
(a) **(b)** **(c)**

Figure 7. For comparison, a simple thresholding of the normalized difference in z coordinates. **(a)** Matches with normalized z differences > 3 units, for the same matches as in Figure 6(c) above. **(b)** Same as (a), for z differences > 6 units (which is the average difference found in Figure 6). **(c)** The union of all differences for all pairs of images, as in Figure 6(d).



(a) **(b)** **(c)**

Figure 8. A second urban scene. **(a)** One of 4 images of the scene taken at time 1. **(b)** One of 3 images taken at time 2. Note the changed building near the center. **(c)** The union of all significant differences found between matches derived from all pairs of images taken at time 1 and all pairs of images taken at time 2.



s

(a) **(b)** **(c)**

Figure 9. A third urban scene. **(a)** One of 4 images of the scene taken at time 1. **(b)** One of 3 images taken at time 2. Note that there are no obvious changes in this case. **(c)** The union of all significant differences found between matches derived from all pairs of images taken at time 1 and all pairs of images taken at time 2. Note that no changes were detected.