Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition
Hilton Head, South Carolina, June 2000 · Volume 2 · Pages 52-57

Maximum-Likelihood Template Matching

Clark F. Olson
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, Mail Stop 125-209, Pasadena, CA 91109

Abstract

In image matching applications such as tracking and stereo matching, it is common to use the sum-of-squared-differences (SSD) measure to determine the best match for an image template. However, this measure is sensitive to outliers and is not robust to template variations. We describe a robust measure and efficient search strategy for template matching with a binary or greyscale template using a maximum-likelihood formulation. In addition to subpixel localization and uncertainty estimation, these techniques allow optimal feature selection based on minimizing the localization uncertainty. We examine the use of these techniques for object recognition, stereo matching, feature selection, and tracking.

1 Introduction

Template matching is a common tool in many applications, including object recognition, stereo matching, and feature tracking. Most applications of template matching use the sum-of-squared-differences (SSD) measure to determine the best match. Unfortunately, this measure is sensitive to outliers and it is not robust to variations in the template, such as those that occur at occluding boundaries in the image. Furthermore, it is important in most applications to know when a match has a significant uncertainty or the possibility exists that a qualitatively incorrect position has been found.

We describe techniques for performing template matching with subpixel localization, uncertainty estimation, and optimal feature selection using a robust measure. In this problem, we search for one or more templates in an image. For example, we may use the features detected in one image as the templates in order to perform tracking in a subsequent image. These techniques are general with respect to the set of pose parameters allowed. We formulate the method using two-dimensional edge and intensity templates with the pose space restricted to translations in the plane in order to simplify the presentation. However, the techniques can be adapted to other problems.

The basic image matching technique that we use is a maximum-likelihood formulation of edge template matching [2] that we have extended to include matching of greyscale templates. In this formulation, a function is generated that assigns a likelihood to each of the possible template positions. For applications in which a single instance of the template appears in the image, such as tracking or stereo matching, we accept the template position with the highest likelihood if the matching uncertainty is below a specified threshold. For other recognition applications, we accept all template positions with likelihood greater than some threshold. A multi-resolution search strategy [1] is used so that not all of the template positions need to be considered explicitly, while still finding the best position in a discretized search space.

Since the likelihood function measures the probability that each position is an instance of the template, error and uncertainty will cause the peak to be spread over some volume of the pose space. Integrating the likelihood function under the peak yields an improved measure of the quality of the peak as a location of the template. We perform subpixel localization and uncertainty estimation by fitting the likelihood surface with a parameterized function at the locations of the peaks. The probability of a qualitative failure is estimated in tracking and stereo matching applications by comparing the integral of the likelihood under the most likely peak to the integral of the likelihoods in the remainder of the pose space. These techniques are also used to perform optimal feature selection, where the features selected for tracking are those with the smallest expected uncertainty.

We demonstrate the utility of these techniques in several experiments, including object recognition through edge template matching, subpixel stereo matching with outlier rejection, and feature selection and tracking in intensity images.

2 Maximum-likelihood matching

Our method is based upon maximum-likelihood edge matching [2], which we describe here and extend to intensity templates.

To formalize the problem, let us say that we have a set of template edge pixels, $M = \{\mu_1, ..., \mu_m\}$, and a set of image edge pixels, $N = \{\nu_1, ..., \nu_n\}$. The elements of M and N are vectors of the x and y image coordinates. We let $p \in T$ be a random variable describing the position of the template in the image. While this makes an implicit assumption that exactly one instance of the model appears in the image, we may set a threshold on the likelihood at each position for cases where the model may not appear or may appear multiple times.

2.1 Map similarity measure

In order to formulate the problem in terms of maximum likelihood estimation, we must have some set of measurements that are a function of the template position in the image. We use the distance from each template pixel (at the position specified by some $p = [x\ y\ ]^t$) to the closest edge pixel in the edge map as this set of measurements. We denote these distances $D_1(p), ..., D_m(p)$. In general, these

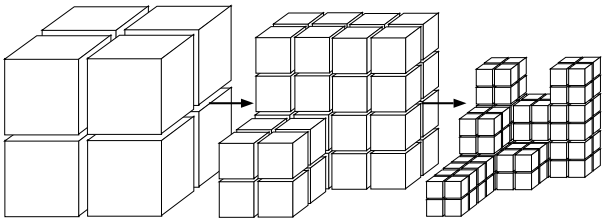


Figure 1: A search strategy is used that recursively divides and prunes cells of the search space.

distances can be found quickly for any p if we precompute the distance transform of the image [5, 7].

We formulate the likelihood function for p as the product of the probability density functions for the distance variables. This make the approximation that the distance measurements are independent. We have found that this yields accurate results, since the correlation between the distances falls off quickly as the points become farther apart.

$$L(p) = \prod_{i=1}^{m} f(D_i(p))$$

This likelihood function is completely independent of the space of template transformations that are allowed. It is defined by the locations to which the template position maps the template edges into the image.

2.2 Search strategy

The search strategy that we use to locate instances of the template in the image is a variation of the multi-resolution technique described by Huttenlocher and Rucklidge [1, 8]. This method divides the space of model positions into rectilinear cells and determines which cells could contain a position satisfying the acceptance criterion. The cells that pass the test are divided into subcells, which are examined recursively. The rest are pruned (Fig. 1). If a conservative test is used, this method is guaranteed to find the best location in a discretized search space.

In order to determine whether some cell C in the pose space may contain a position meeting the criterion, we examine the pose c at the center of the cell. A bound is computed on the maximum distance between the location to which an edge pixel in the template is mapped by c and by any other pose in the cell. We denote this distance Δ_C. If we treat template positions as functions that map template pixels into the image then we can write Δ_C as follows:

$$\Delta_C = \max_{p \in C} \max_{m \in M} ||p(m) - c(m)||$$

Now, to place a bound on the quality of the cell, we compute a bound on the minimum distance from each edge pixel in the template to any edge pixel in the image that can be achieved over the entire cell. This is done by subtracting the maximum change over the cell, Δ_C, from the distance achieved at the center of the cell, $D_i(c)$:

$$D_i^C = \max(D_i(c) - \Delta_C, 0)$$

Propagating these values through the probability density function yields a bound on the likelihood score that can be achieved by any position in the cell:

$$\max_{p \in C} L(p) \leq \prod_{i=1}^{n} f(D_i^C)$$

If this bound does not surpass the best that we have found so far (or some threshold, if we seek multiple instances), then the entire cell is pruned from the search. Otherwise, the cell is divided into two cells by slicing it along the longest axis and the process is repeated recursively on the subcells. In practice, the pose space is discretized at pixel resolution and the recursion ends when a cell is reached that contains a single pose in the discretized space, which is tested explicitly.

2.3 Greyscale templates

While these techniques have, so far, been described in terms of binary edge maps, they can be extended to greyscale templates by considering the image to be a surface in three dimensions (x, y, and intensity). We will thus describe the techniques in terms of *occupied pixels*, which are the edges in an edge map or the intensity surface in the three-dimensional representation of a greyscale image. The templates and images can thus be considered to be sets of 3-vectors, corresponding to the occupied pixels. We must now define a distance function over the three dimensions for greyscale images and compute nearest neighbors with respect to this distance, but the remainder of the method is unchanged.

For two pixels $\mu_i = (x_i, y_i, z_i)$ and $\nu_j = (x_j, y_j, z_j)$, where z is the intensity, we have used a variation of the L_1 distance metric, since this makes the distance computations simple:

$$D(\mu_i, \nu_j) = |x_i - x_j| + |y_i - y_j| + \gamma |z_i - z_j|$$

The value of γ should be chosen such that the errors in each dimension have the same standard deviation.

3 Estimating the PDF

For the uncertainty estimation to be accurate, it is important that we use a probability density function (PDF) that closely models the sensor uncertainty. In this past we have used a robust (but heuristic) measure [2]. We develop a similar measure here using the principle that the density can be modeled as the sum of two terms (one for inliers and one for outliers):

$$f(d) = \alpha f_1(d) + (1 - \alpha) f_2(d)$$

The first term describes the error distribution when the pixel is an inlier (in the sense that the location that generated the template pixel also appears in the image). Usually, we can model this distribution as normal in the distance to the closest occupied pixel. For the 2D case, this yields:

$$f_1(d) = \frac{1}{2\pi\sigma^2} e^{-(d_x^2 + d_y^2)/2\sigma^2} = \frac{1}{2\pi\sigma^2} e^{-d^2/2\sigma^2}$$

Note that this is a bivariate probability density in (d_x, d_y), rather than a univariate probability density in $||d||$, which would imply rather different assumptions about the error distribution. Formally, we should think of d as a 2-vector of the x and y distances to the closest occupied pixel in the image. However, to compute the probability density function, it will only be necessary to know the magnitude of d. Thus, the orientation of the distance vector is irrelevant.

For greyscale image matching, we use:

$$f_1(d) = \frac{1}{(2\pi\sigma^2)^{\frac{3}{2}}} e^{-d^2/2\sigma^2}$$

While the distance measure that we use for greyscale images is not Euclidean, it has resulted in excellent results. Alternatively, we could use Euclidean distances with a more complex distance transform algorithm.

The second term in the PDF describes the error distribution when the cell is an outlier. In this case, the template pixel does not appear in the image for some reason (such as occlusion). In theory, this term should also decrease as d increases, since even true outliers are likely to be near some occupied pixel in the image. However, this allows pathological cases to have an undue effect on the likelihood for a particular template position. In practice, we have found that modeling this term as the expected probability density for a random outlier yields excellent results.

$$f_2(d) = f_{exp}$$

It should be noted that $f_2(d)$ is not a probability distribution, since it does not integrate to unity. This is unavoidable in a robust measure, since any true probability distribution must become arbitrarily close to zero for large values of $D_i(t)$.

It is interesting to note that we could achieve the same results as the SSD measure by assuming that there are no outliers ($\alpha = 1$) and using:

$$D(\mu_i, \nu_j) = \begin{cases} z_i - z_j, & \text{if } x_i = x_j \text{ and } y_i = y_j \\ \infty, & \text{otherwise.} \end{cases}$$

The maximum-likelihood measure gains robustness by explicitly modeling the possibility of outliers and allowing matches against pixels that do not precisely overlap the template pixel.

Let us now consider the constants in this probability density function. First, α is the probability that any particular occupied pixel in the template is an inlier in the image. We must estimate this value based on prior knowledge of the problem and thus it is possible that we may use an inaccurate estimate of this value. However, we have found that the localization is insensitive to the value of this variable. Next, σ is the standard deviation of the measurements that are inliers. This value can be estimated by modeling the characteristics of the sensor or it can be estimated empirically by examining real data, which is the method that we have used in our experiments. Finally, f_{exp} is the expected probability density for a random outlier point. Recall that we use a bivariate probability density function for edge matching. For this case, we have:

$$f_{exp} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(d_x, d_y)^2 dx dy$$

This value can be estimated efficiently using the Euclidean distance transform of the image. We first compute a histogram of the signed x and y distances to the nearest neighbor for every pixel in the image. These values can be computed easily as a by-product of the computation of the distance transform [5]. For an image with $W \times H$ pixels and distance transform histogram $h(x, y)$, we can approximate f_{exp} as:

$$f_{exp} \approx \sum_{x=-W}^{W} \sum_{y=-H}^{H} \frac{h(x,y)}{WH}^2$$

This can also be extended to the greyscale case. In practice, the use of an empirical estimate does not have a large effect on the matching results.

4 Subpixel localization

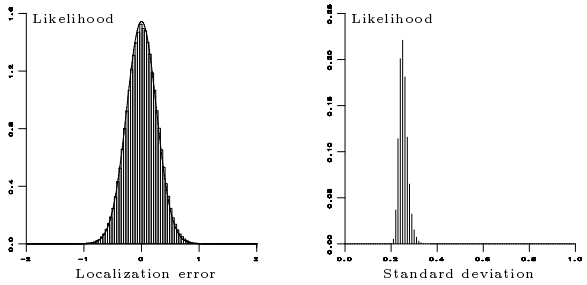
With the probabilistic formulation of template matching described above, we can estimate the uncertainty in the localization in terms of both the variance of the estimated positions and the probability that a qualitative failure has occurred. Since the likelihood function measures the probability that each position is the actual model position, the uncertainty in the localization is measured by the rate at which the likelihood function falls off from the peak. In addition, we perform subpixel localization in the discretized pose space by fitting a function to the peak that occurs at the most likely model position.

Let us take as an assumption that the likelihood function approximates a normal distribution in the neighborhood around the peak location. Fitting such a normal distribution to the computed likelihoods yields both an estimated variance in the localization estimate and a subpixel estimate of the peak location. While the approximation of the likelihood function as a normal distribution may not always be precise, it yields a good fit to the local neighborhood around the peak and our experimental results indicate that accurate results are achieved with this approximation.

Now, we perform our computations in the domain of the logarithm of the likelihood function:

$$\ln L(x, y) = \sum_{i=1}^{m} \ln f(D_i(p))$$

Since the logarithm of a normal distribution is a polynomial of order 2, we fit the peak in the log-likelihood function with such a polynomial. For simplicity, let us assume



(a) (b)

Figure 2: Distribution of errors and estimated standard deviations in synthetic template matching experiment. (a) Comparison of estimated distribution of localization errors (solid line) to observed distribution (bar graph). (b) Distribution of estimated standard deviations.

independence in the errors in x and y. (This is unnecessary, but simplifies the presentation.) In this case, we have:

$$\ln L(x,y) \approx \ln \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{(x-x_0)^2}{2\sigma_x^2} - \frac{(y-y_0)^2}{2\sigma_y^2}}$$

$$= -\frac{(x-x_0)^2}{2\sigma_x^2} - \frac{(y-y_0)^2}{2\sigma_y^2} + \ln \frac{1}{2\pi\sigma_x\sigma_y}$$

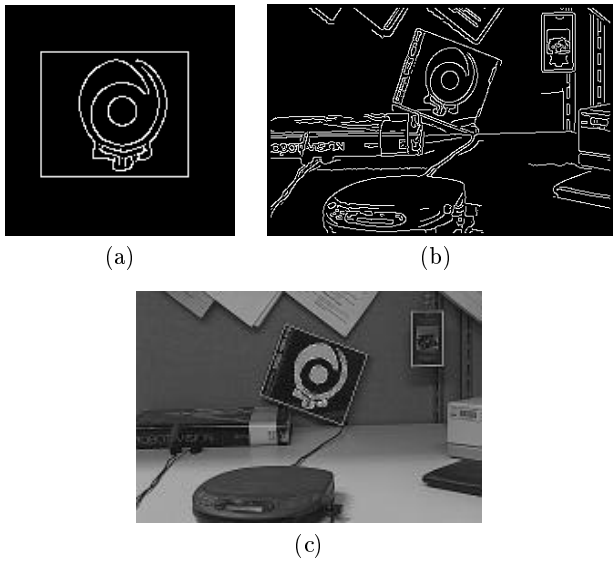
Fitting this polynomial using a least-squares error criterion is straightforward [6]. The values of x_0 and y_0 yield the subpixel localization result, since this is the estimated location of the peak in the likelihood function. In addition, σ_x and σ_y yield estimates for the uncertainty in the localization result. These results can be extended to similarity transformations or full affine transformations with only minor modifications.

5 Probability of failure

In addition to estimating the uncertainty in the localization estimate, we can use the likelihood scores to estimate the probability of a failure to detect the correct position of the template. We address the case where exactly one instance of the template occurs in the image, such as in tracking or stereo matching.

We estimate the probability of failure by summing the likelihood scores under the peak selected as the most likely model position and comparing to the sum of the likelihood scores that are not part of this peak. In practice, we can usually estimate the sum under the peak by examining a small number of values around the peak, since they fall off quickly.

The values for the remainder of the pose space can be estimated efficiently with some additional computation during the search. Whenever a cell in the search space is considered, we compute not only a bound on the maximum score that can be achieved, but also an estimate on the average score that is achieved by determining the score for the center of the cell. If the cell is pruned, then the sum is incremented by the estimated score multiplied by the size of the cell. In practice, this yields a good estimate, since regions with large scores cannot be pruned until the cells become small. We thus get good estimates when the score is large and, when the estimate is not as good, it is because the score is small and does not significantly affect the overall sum.



(a) (b)

(c)

Figure 3: Object recognition example. (a) Edge template. (b) Edges extracted. (c) Recognition result.

Let S_p be the sum obtained for the largest peak in the pose space and S_n be the sum for the rest of the pose space. We can estimate the probability of correctness for the largest peak as:

$$P_c = \frac{S_p}{S_p + S_n}$$

6 Experiments with synthetic data

These techniques were first tested on synthetic data for which we could compare the performance of the techniques with precise ground truth. In these experiments, we randomly generated synthetic templates containing 60 feature points in a 64×64 unit square. An image containing half of these points was then generated using a random translation and with a Gaussian error with standard deviation $\sigma = 1.0$ pixels. In addition, 530 clutter edge points were added such that the density of edges was the same at the position of the model as in the rest of the image. Detection of the template was then performed using the techniques described above. Over 100,000 trials, the template was correctly detected in all but 2 of the cases, with an average error in the correct trials of 0.211 pixels in each dimension. The average estimated standard deviation in the localization using the techniques from the previous section was 0.258 units.

Figure 2(a) shows the distribution of actual errors observed versus the distribution that we expect from the average standard deviation estimated in the trials. The close similarity of the plots indicates that the average estimated standard deviation is a good estimate of the actual value. This also validates the approximation of the likelihood function as a normal distribution in the neighborhood of the peak. Figure 2(b) shows the distribution

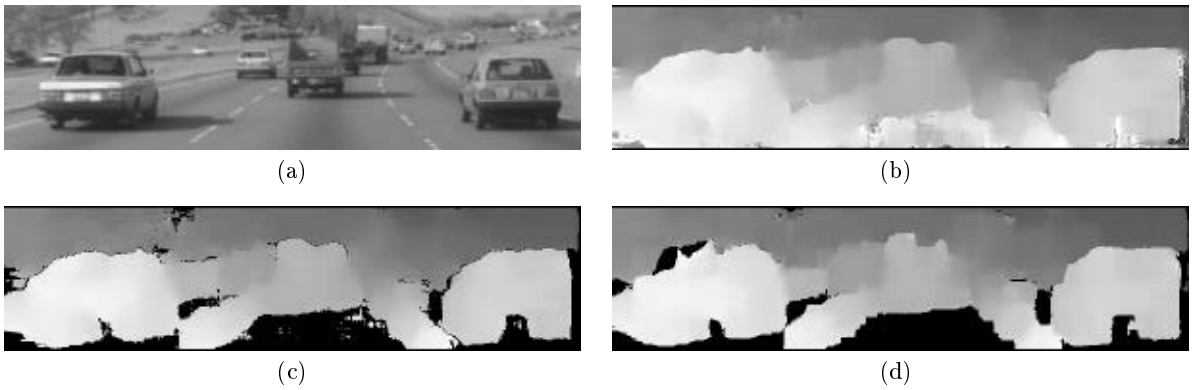


Figure 4: Stereo matching example. (a) Left image of stereo pair. (b) Complete disparity map computed. (c) Disparity map after pruning. (d) Disparity map computed using SSD.

of the estimated standard deviations in this experiment. It can be observed that the estimate is consistent between trials, since the plot is strongly peaked near the location of the average estimate. Taken together, these plots indicate that the standard deviation estimates are likely to be accurate for each individual trial.

When compared to a version of these techniques that does not perform subpixel localization, our method reduces the error in the localization by 33.7%. These techniques thus improve both the localization result significantly and yield accurate estimates of the standard deviation of the localization result.

7 Applications

These techniques have been tested in applications including object recognition, stereo matching, and feature selection and tracking. The application of similar techniques to mobile robot localization has also been explored in [3, 4].

7.1 Object recognition

These techniques can be applied to object recognition using greyscale or edge templates. Given a model template, we search over some space of transformations to locate the template in the image. In this case, we considered similarity transformations of an edge template. Figure 3 shows a simple example where the use of subpixel localization yields an improvement in the pose estimate. The rotation of the template was discretized in 5° intervals. When subpixel localization was not used, the detected rotation of the template was inaccurate, with an error of approximately 3°. This problem is easily corrected when subpixel localization is used.

We note that a finer discretization of the pose space might yield comparable results to the subpixel localization method. However, this would require more computation. The subpixel localization techniques can thus be viewed as a technique to improve the computation time of matching rather than the precision.

7.2 Stereo matching

These techniques can be applied to stereo matching in a fashion similar to correlation-based methods, where small windows from one image are used as the templates, and they are matched against windows in the other image.

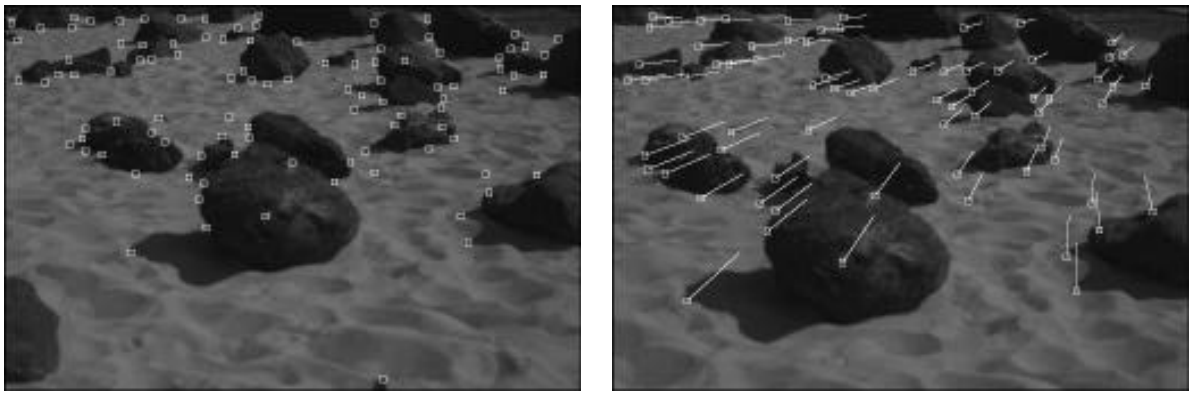
Figure 4 shows an example where stereo matching was performed using these techniques. The disparity, uncertainty, and probability of a false match were computed for each template in the left image of the stereo pair by matching against the right image. Figure 4(b) shows the complete disparity map, which contains outliers and inaccuracies due to occlusions and image regions with low texture. Figure 4(c) displays the disparity map after pruning the locations for which the uncertainty estimate or probability of failure is large. No outliers remain in the disparity map.

For comparison, Figure 4(d) shows the result of applying these same techniques using the SSD measure. For this case, we can still compute an uncertainty and probability of failure using our techniques. However, the results are less accurate. A small number of outliers remain in the disparity map. In addition, this measure yields lower quality results in the neighborhoods of occlusion boundaries, since it is less robust to changes in the composition of the template.

7.3 Feature selection and tracking

Since our techniques provide estimates of the uncertainty for matching a template, they can be easily adapted to perform feature selection for tracking with optimal matching uncertainty. This is performed by estimating the uncertainty of matching each possible feature with the region in the image in which it lies.

We first compute a distribution that captures the probability of each image intensity at the potential template locations. This distribution models only the changes in pixel intensity as the camera moves. The translation of the pixels in the image is ignored here, since the template matching searches over the translations. To estimate this distribution, we initially take the intensity surface of the template to have probability 1. This distribution is



(a) (b)

Figure 5: Optimal feature selection for greyscale matching. (a) Features selected. (b) Tracked features.

smoothed in both position and intensity to model noise and warping as the camera moves. We then perform uncertainty estimation for each possible template by matching against the computed distribution (which is treated as a three-dimensional image). The locations with the lowest uncertainty are selected as the optimal features to track.

Figure 5 shows an example of the feature selection techniques applied to an image of rocky terrain. In this case, 100 7×7 feature templates were selected as having the lowest uncertainty for tracking. We then performed tracking in a subsequent image, after the camera had undergone forward motion. For each selected feature, we searched the entire post-move image for a match, although, in practice, the search space would usually be limited to a smaller region. Figure 5(b) shows the 72 features that survived pruning using the uncertainty and probability of failure measures. No false positives remain in the tracked features.

To compare against SSD matching, this same procedure was applied to a sequence of images similar to Figure 5. Over this set of images, our feature selection and tracking techniques tracked 70.6% of the features, with 1.6% outliers. For the same images, SSD matching techniques tracked only 38.0% with 2.3% outliers. Thus, even with a lower tracking rate, the SSD techniques yield a higher rate of outliers owing to the lower robustness to occluding boundaries and intensity variations.

8 Summary

Template matching techniques using the SSD measure are susceptible to errors in the presence of outliers and occlusion boundaries. In addition, it is important in many applications to perform accurate subpixel localization and uncertainty estimation. This work has developed methods to perform these tasks using robust template matching for greyscale and edge templates. The basic matching framework that we use is maximum-likelihood estimation of the template position in the image. In order to perform subpixel localization and uncertainty estimation, we fit the peak in the likelihood function with a normal distribution. The summit of the distribution yields the localization estimation and the standard deviation of the distribution yields an estimate on the uncertainty of the localization. The probability of a qualitative failure is estimated by examining the likelihood scores. Experiments on synthetic data have confirmed that these techniques yield improved localization results and accurate uncertainty estimates. The use of these techniques in several applications has yielded excellent results.

Acknowledgements

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

[1] D. P. Huttenlocher and W. J. Rucklidge. A multi-resolution technique for comparing images using the Hausdorff distance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 705–706, 1993.

[2] C. F. Olson. A probabilistic formulation for Hausdorff matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 150–156, 1998.

[3] C. F. Olson. Subpixel localization and uncertainty estimation using occupancy grids. In *Proceedings of the International Conference on Robotics and Automation*, volume 3, pages 1987–1992, 1999.

[4] C. F. Olson and L. H. Matthies. Maximum-likelihood rover localization by matching range maps. In *Proceedings of the International Conference on Robotics and Automation*, volume 1, pages 272–277, 1998.

[5] S. Pavel and S. G. Akl. Efficient algorithms for the Euclidean distance transform. *Parallel Processing Letters*, 5(2):205–212, 1995.

[6] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Plannery. *Numererical Recipes in C*. Cambridge University Press, 1988.

[7] A. Rosenfeld and J. Pfaltz. Sequential operations in digital picture processing. *Journal of the ACM*, 13:471–494, 1966.

[8] W. J. Rucklidge. *Efficient Visual Recognition Using the Hausdorff Distance*. Springer-Verlag, 1996.