MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY
and
CENTER FOR BIOLOGICAL AND COMPUTATIONAL LEARNING
DEPARTMENT OF BRAIN AND COGNITIVE SCIENCES

A.I. Memo No. 1510 December, 1994
C.B.C.L. Memo No. 109

Fast Object Recognition in Noisy Images Using Simulated Annealing

Margrit Betke **Nicholas C. Makris**

This publication can be retrieved by anonymous ftp to publications.ai.mit.edu.

Abstract

A fast simulated annealing algorithm is developed for automatic object recognition. The object recognition problem is addressed as the problem of best describing a match between a hypothesized object and an image. The normalized correlation coefficient is used as a measure of the match. Templates are generated on-line during the search by transforming model images. Simulated annealing reduces the search time by orders of magnitude with respect to an exhaustive search. The algorithm is applied to the problem of how landmarks, for example, traffic signs, can be recognized by an autonomous vehicle or a navigating robot. Images are assumed to be taken while the robot or the vehicle is moving through its environment. It tries to match them with templates created online from models stored in a database. We illustrate the performance of our algorithm with real-world images of complicated scenes with traffic signs. False positive matches occur only for templates with very small information content. To avoid false positive matches, we propose a method to select model images for robust object recognition by measuring the information content of the model images. The algorithm works well in noisy images for model images with high information content.

This report describes research done at the Center for Biological and Computational Learning and the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the Center is provided in part by a grant from the National Science Foundation under contract ASC–9217041.

The first author can be reached at margrit@ai.mit.edu. The second author can be reached at Naval Research Laboratory, Washington, D.C. 20375.

1 Introduction

The field of automated object recognition is one of the most complex areas in computer vision and image understanding. Object recognition based on matched filtering has been a very active research area in computer vision for many years. Matched filtering has been used much earlier in the areas of radar, sonar, and signal processing [Opp78]. Valuable information for visual object recognition can be obtained from that literature.

Although template matching has been widely used in computer vision [BB82, Yar85], a crucial problem with the method is the size of the search space [MR90, LC88]. There are several approaches published in the literature that either reduce the size of the search space or that direct the search towards areas in the search space for which a match is more likely [Gri88, Gri90, NR72, MR90, AF86]. In this paper a new approach is proposed that uses both such techniques. We discuss the problem of how certain landmarks, for example traffic signs, can be recognized by an autonomous vehicle or robot. For this particular application, a five-dimensional search-space is sufficiently large for robust object recognition and small enough for efficient object recognition.

The method presented constructs templates *on-line* during the search. The algorithm uses an efficient local definition of the correlation coefficient to evaluate the match. The algorithm presented correctly finds the location, shape, size, and orientation of objects. If enough independent information is contained in a template image, it can be matched with an object in an image uniquely. False positive matches occur only for objects that have very small information content. To avoid false matches, templates with insufficient information content should not be used for recognition tasks. We describe how to compute the information content of template images.

Although the main objective of this paper is to describe a new approach to the general problem of visual object recognition, the solution to the special problem of recognizing traffic signs is significant by itself. Automatically recognizing traffic signs in images is very valuable for mobile robot or autonomous vehicle navigation. A robot that can recognize a traffic sign as a familiar landmark in its map of the environment can then use this information to localize itself in its environment [BG94, Bra90]. Our method stands apart from previous approaches to traffic sign recognition because first, it is efficiently applied to real-world landscape images (as opposed to Ettinger's isolated signs [Ett88]), and second, it does not rely on color perception which is very sensitive to lighting changes. This sensitivity limits the approach of May [May94] and Zheng et al. [ZRJ94] who address the problem of recognizing traffic signs using color information.

The optimization technique *fast simulated annealing* is applied to avoid the cost of brute-force search by directing the search successfully. It reduces the search time by orders of magnitude. Recent publications in the sonar literature [CBK+93, KCPD90] show that fast simulated annealing has been very successful in coherent signal extraction and localization in noisy environments. We use it in a similar way for incoherent image processing. Kirkpatrick et al. [KGV83] show how to implement a Metropolis algorithm [MRR+53] to simulate annealing of combinatorial optimization problems. Szu and Hartley [SH87] propose an inverse linear cooling schedule for simulated annealing. This version is called "fast simulated annealing." The original slower version of simulated annealing has been applied to segmentation and noise reduction of degraded images by Geman and Geman [GG84], to represent lobed objects by Friedland and Rosenfeld [FR91], and to boundary detection by Geman et al. [GGGD90]. However, for visual object recognition, fast simulated annealing has yet not been exploited.

This paper is organized in the following way: The object recognition problem is defined as a parameter search problem in Section 2. Section 3 shows how templates are generated from model images. Section 4 examines the search space of the recognition problem and introduces "ambiguity surfaces." Section 5 describes our simulated annealing algorithm and Section 6 reports our experimental results. Section 7 analyzes the error in the correlation and proposes how to avoid false matches. Section 8 describes our results on noisy images. We conclude with a summary of this work and suggestions how to apply these results to other problems.

2 The Recognition Problem

An object in an image I is defined to be recognized if it correlates highly with a template image T of the hypothesized object. This template image T is a transformed version of the model of the hypothesized object. Model images of objects are stored in a library. Section 3 shows how to compute the template from the model. A template $T(x,y)$, for $0 \le x < n_T, 0 \le y < m_T$, is generally much smaller than the image $I(x,y)$. The template is compared with the part $I_T(x,y)$ of image $I(x,y)$ that contains the hypothesized object. Assuming pixel (x_0, y_0) is at the lower-left corner of the hypothesized object in I, subimage I_T is defined to be

$$I_T(x,y) = I(x_0+x, y_0+y) \quad \text{for } 0 \le x < n_T, 0 \le y < m_T.$$

We use the normalized correlation coefficient as a measure of how well images I_T and T correlate or match. For images I_T and T, the normalized correlation coefficient ρ is the covariance of I_T and T normalized by the standard deviation of I_T and T. The correlation coefficient is dimensionless, and $|\rho| \le 1$. The correlation coefficient measures how accurate image I_T can be approximated by template T. Image I_T and template T are perfectly correlated if $\rho = 1$. We approximate ρ using the sampled coefficient of correlation

$$r = \Big(p_T \sum_{x,y} I_T(x,y)T(x,y) - \Big(\sum_{x,y} I_T(x,y)\Big) \cdot \Big(\sum_{x,y} T(x,y)\Big)\Big)/\sigma_{I_T}\sigma_T$$

where $\sigma_{I_T} = \sqrt{p_T \sum_{x,y} I_T(x,y)^2 - \Big(\sum_{x,y} I_T(x,y)\Big)^2}$, $\sigma_T = \sqrt{p_T \sum_{x,y} T(x,y)^2 - \Big(\sum_{x,y} T(x,y)\Big)^2}$ and p_T is the number of pixels in the template image T with nonzero brightness values and $p_T \le n_T \cdot m_T$. Note this last condition means that not all the pixels in images T and I_T are actually compared but only the nonzero pixels in T with the corresponding pixels in I_T. This is important, for example, if the template contains a circular object. Here pixels in T bordering the circle (or the background) will be zero (black). The computation time of r is proportional to the number of pixels in the hypothesized object, which is usually much smaller than the number of pixels in I. Using the correlation as a measure of successful recognition is also advantageous because it is a very robust measure. That is, it is relatively insensitive to fluctuations in the environment compared to higher resolution methods, as is well documented in spectral, bearing, and range estimation problems [Joh82, BKM93].

3 Generating Templates from Model Images

A template $T(x, y)$ is generated from a model image $M(x, y)$ by choosing three parameters that describe a transformation from M into T. The parameters determine how the model is sampled, and if necessary, how it is interpolated to generate the template. The parameters used are a rotation parameter τ and two sampling parameters s_x and s_y.

For notational convenience, we define the origin of a coordinate system for model image $M(x, y)$ to be in the middle of the image, i.e., $M(x, y)$ is defined for $-(n_M - 1)/2 \leq x \leq (n_M - 1)/2$ and $-(m_M - 1)/2 \leq y \leq (m_M - 1)/2$ for n_M, m_M odd. Then the rotation parameter τ determines how the x and y axes of $M(x, y)$ are rotated to define the x and y axes of $T(x, y)$. More precisely, given vectors

$$\mathbf{m_x} = \left(\frac{n_M - 1}{2}, 0\right) \quad \text{and} \quad \mathbf{m_y} = \left(0, \frac{m_M - 1}{2}\right),$$

which lie on the coordinate axes of M, and model radius $R_M = \sqrt{\left(\frac{n_M-1}{2}\right)^2 + \left(\frac{m_M-1}{2}\right)^2}$, we compute vectors

$$\mathbf{t_x} = R_M\,(\cos\tau, \sin\tau) \quad \text{and} \quad \mathbf{t_y} = R_M\,(-\sin\tau, \cos\tau)$$

which define the coordinate axes of the template image T in continuous space. The axes of T always span the model object as show in Figure 1.

The sampling parameters s_x and s_y determine how many samples along vectors $\mathbf{t_x}$ and $\mathbf{t_y}$ are used for the template image, respectively. The spacing between the samples along $\mathbf{t_x}$ is $((n_M - 1)/2)/s_x$. If there is a pixel in $M(x, y)$ after every $(n_M - 1)/(2s_x)$ step along $\mathbf{t_x}$, its brightness is used to define T along its x-axis. For example this scenario may occur if $\tau = 45$ degrees, and $s_x = (n_M - 1)/2$. As shown in Figure 1, if $s_x = (n_M - 1)/4$ the model is down-sampled and transformed into a template that is about one-quarter the size of the model. Pixels of zero brightness are added where necessary as shown in Figure 1.

In general, there may not be a pixel in M at the sampling point on vector $\mathbf{t_x}$. If this is the case, we use a four-point interpolation to define the brightness for the template at that point. Similarly, M is sampled (and if necessary interpolated) along vectors $\mathbf{t_y}, -\mathbf{t_x}$, and $-\mathbf{t_y}$ to obtain the brightness of the template pixels along the template coordinate axes. The rest of the template is now determined from M along the grid that is defined by the samples on the template coordinate axes.

Since the sampling rates s_x and s_y in the template coordinate system are different in general, the template is a rotated, scaled, and uniformly deformed version of the model. More parameters would be needed to describe more general non-uniform and non-linear deformations of the model. A straightforward extension would be to add a fourth parameter to obtain a non-uniform linear deformation of the model. However, for our purposes, the transformation described is sufficient because the objects to be recognized are usually flat, normal to the viewing direction and far away from the camera compared to the object size. Our method computes the template very quickly by sweeping over the model image only once. The time for creating a $n_T \times m_T$ template image is $O(n_T m_T)$.

Examples of a model and corresponding transformed templates are shown in Figure 2. The first two templates are scaled by $s_x = s_y$ and are not rotated. The remaining templates in Figure 2 are defined by more general transformations with $s_x \neq s_y$.

4 The Parameter Search Space

The space of possible solutions of the recognition problem is extremely large, even if a particular object is known to be in the image a priori. The dimension of the search space is determined by the number of possibilities for position, size, shape, and orientation of the object. The number of possibilities for the position of the centroid of the object in the image is $O(n^2)$ for a $n \times n$ image. Assuming that the size and shape of the object can be approximated by sampling the model along two perpendicular axes as described in the previous section, the number of possibilities to approximate the size and shape of the object is also $O(n^2)$. Even with this assumption, the number of possible angles is still very large; since the image is discrete, we assume that the number of possible angles is $O(n)$. Thus, the size of the search space is $O(n^5)$ for an $n \times n$ image. For a typical image of size 256×256, the search space has a size of order 10^{14}. An exhaustive search of this space would take too long to find a good match between templates and images.

We use terminology from the radar and sonar literature to describe the search space. We call the space an *ambiguity surface*. A peak in the ambiguity surface means that the correlation coefficient is high for a particular set of parameters. Figure 3 shows an example of a two-dimensional ambiguity surface with a peak shown in black. There may be several peaks in an ambiguity surface. If the template and the object in the image match perfectly, the cross-correlation between template and image results in a peak in the ambiguity surface which is the global optimum. Due to noise and reduction of the search space by our template transformation, we do not expect a perfect match. However, in most cases the global optimum corresponds to a correct match or

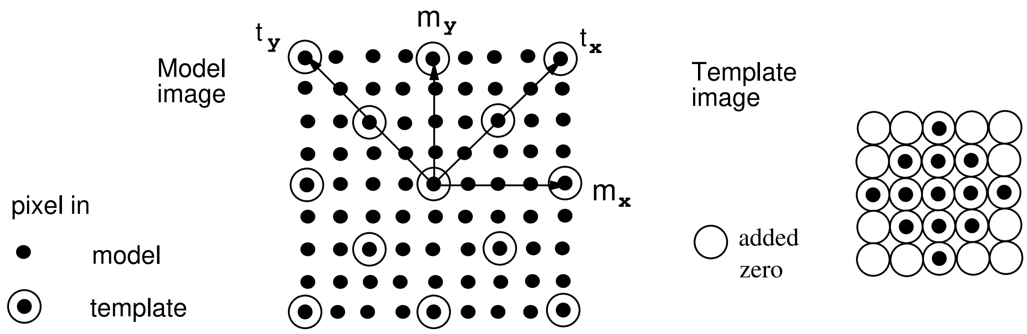

Figure 1: A 5×5 template image is obtained from a 9×9 model image using parameters $s_x = s_y = 2$ and $\tau = 45$ degrees.

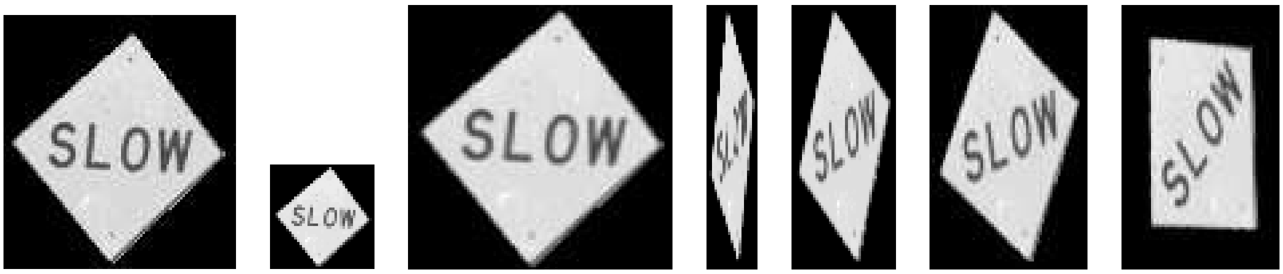

Figure 2: Model of slow sign with 101×111 pixels, and six templates of slow sign. Templates are obtained by sampling model sign at various sampling rates and degrees of rotation.

Figure 3: On the left, image Slow3. On the right, the ambiguity surface of image Slow3 computed for all possible translations given fixed angle and scaling parameters. A deterministic search would compute each value on this surface. A steepest descent procedure would fail because of local minima. Therefore, a stochastic search is used to find the best correlation value (here the darkest pixel value).

recognition.

As we can also see in Figure 3, an iterative search for a peak in the ambiguity surface such as steepest descent would fail because it would get "stuck" in local minima. Simulated annealing, however, is able to "jump" out of local minima and find the globally best correlation value.

5 The Simulated Annealing Algorithm

In this section we describe our algorithm for finding an optimal match between images and templates. Our algorithm is based on a fast version of simulated annealing. Simulated annealing has become a popular search technique for solving optimization problems. Its name originates from the process of slowly cooling molecules to form a perfect crystal. The cooling process and its analogous search algorithm is an iterative process, controlled by a decreasing temperature parameter. At each iteration, our algorithm generates templates on-line as described in Section 3. New test values for the location, sampling, and rotation parameters of the template are randomly perturbed from current values. If the correlation coefficient r_j increases over the previous coefficient r_{j-1}, the new parameter values are accepted in the j-th iteration (as in the gradient method). Otherwise, they are accepted if

$$e^{-(E_j - E_{j-1})/T_j} > \xi$$

where ξ is randomly chosen to be in $[0, 1]$, T_j is the temperature parameter, and $E_j = 1 - r_j$ is the cost function in the j-th iteration. For a sufficient temperature this allows "jumps" out of local minima. We choose

$$T_j = T_0/j \qquad\qquad 1 \leq j \leq L$$

as the cooling schedule for the j-th update of the temperature parameter where T_0 is the initial temperature and L is the number of iterations during the search. Note that the rate at which the temperature decreases is inverse linear as first proposed by Szu and Hartley [SH87] and converges faster than an often used logarithmically inverse cooling schedule [GG84]. As a criteria for stopping the annealing process, we simply put a limit on the search length L. Although this does not ensure convergence to the optimal correlation coefficient, the solutions we obtain for the parameters are generally sufficient and solve the recognition task.

As Kuperman et al. [KCPD90] point out, if the search problem involves different kinds of parameters the annealing algorithm is rather analogous to the cooling of a mixture of liquids, each of which have different freezing points. An algorithm that randomly perturbs all parameters at the same time has poor convergence properties. Therefore, at a specific temperature we do not combine the test for the choice of the location, sampling, and rotation angle. We also obtain good results using simulated annealing only for the location parameters, and a gradient descent procedure [CBK+93] for the remaining parameters given large enough perturbations.

To properly deal with image boundaries of an image $I(x, y)$ for which $0 \leq x < n_I$ and $0 \leq y < m_I$, we use the following formula to perturb the x-coordinate c_x of the centroid position of a template with radius R_T in image $I(x, y)$

$$c_x = \begin{cases} c_x & \text{if } c_x - R_T \geq 0 \text{ and } c_x + R_T \leq n_I \\ -c_x & \text{if } c_x + R_T < 0 \text{ and } c_x - R_T \geq -n_I \\ 2n_I - c_x & \text{if } c_x - R_T > n_I \text{ and } c_x + R_T \leq 2n_I \\ n_I/2 & \text{otherwise (unlikely perturbation).} \end{cases}$$

The y-coordinate c_y of the centroid of the template is perturbed similarly. This formula avoids attracting the centroid position to the rim or corners of the image.

6 Experimental Results

The algorithm described above was implemented on a Sun workstation and on a Silicon Graphics Iris. We used the model images shown in Figure 4 to find templates that correlate optimally with the scene images shown in Figure 5. The images are quantized using 256 grey levels. The size of the model images is 122×117 pixels (except for the one-way sign, which has 178×60 pixels.) The size of the scene images varies between 100×70 and 516×365 pixels.

For all scene images, the shape, size, orientation, and location of any traffic sign is found if it is known a priori what kind of sign to look for. For example, using the stop sign model shown in Figure 4 the algorithm finds the stop sign in a complicated scene image like image Stop5. (This is the second image in the last row of images in Figure 5; see also Figure 6). The stop sign in scene image Stop5 is recognized although the stop sign model was constructed from a picture of a completely different stop sign. Note that the stop sign in image Stop5 has graffiti, while the model sign does not.

For the more general problem of recognizing *which* object is in a scene image (i.e., not knowing the kind of traffic sign a priori), we ran 144 experiments with 18 scene images and 8 model images. Table 1 contains the correlation values obtained in the experiments. For each scene image, our algorithm computes the highest correlation coefficient among the set of values obtained for each model (boldface values in Table 1). The model corresponding to the maximum correlation value is selected as the sign recognized in the scene image. For most scene images, the correlation coefficient is highest if a match between a sign in the image and its corresponding template occurs. Only for three images, Slow2, Stop4, and Stop5, a false positive match occurs because the best correlation coefficient is not the one for the corresponding model. We show the templates causing these false positive matches in Figure 6.

There are two facts that contribute to the false positive matches. First, some models do not have enough structure by themselves and match easily with arbitrary parts of the images. For example, the European no-entry sign's white middle bar matches with the roof of a car in image Stop5, as shown in Image 5 of Figure 6. In Section 7 we analyze this problem quantitatively. Second, some models look quite different from the actual landmark in the scene image. For example, as mentioned before, the stop sign model does not have any graffiti while the signs in Stop4 and Stop5 do. The templates constructed from the model stop sign do not match the

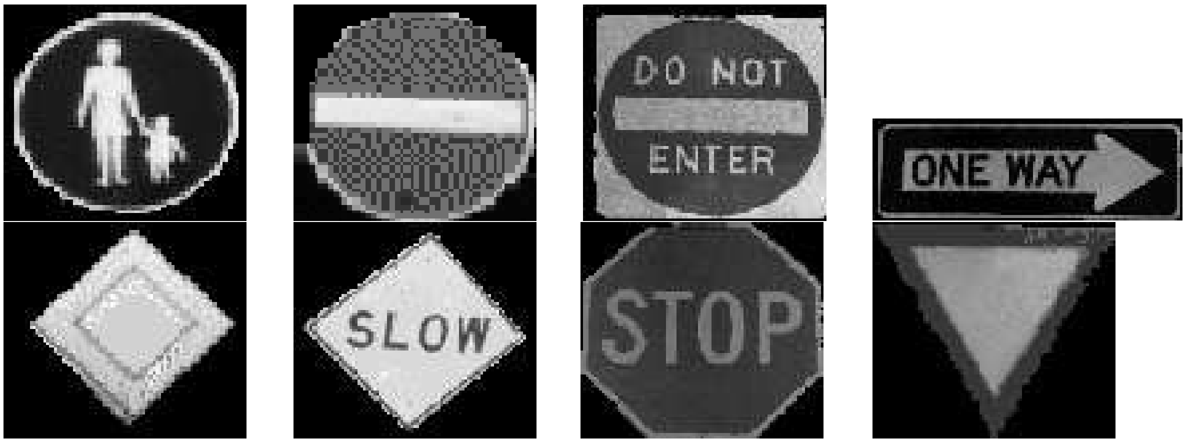

Figure 4: Model images used in experiments: Footpath, E-no-entry, No-entry, One-way, Priority, Slow, Stop, and Yield.

stop signs in images Stop4 and Stop5 well enough to result in a correlation coefficient larger than the one obtained with the model E-no-entry (see Image 4 and 5 of Figure 6). One could try to solve this problem by making a model of each traffic sign (including its graffiti) in the environment. However, this would result in a huge library of signs which would increase the search time substantially. Moreover, the environment may change and outdate the library quickly. Therefore, we instead propose to select a small number of model images with *high information content* (see Section 7) so that false positive matches are avoided.

6.1 Illumination Changes

The correlation coefficient $\rho(I_T, T)$ measures not only how accurate image I_T can be approximated by template T, but also how accurate image I_T can be approximated by a linear function of T, since $\rho(I_T, T) = \rho(I_T, aT + b)$ for some constants a, b. Therefore, the correlation coefficient is invariant to constant scale factors in brightness. Thus recognition is not affected by new lighting conditions that mainly result in such brightness changes.

6.2 Simulated Annealing vs. Exhaustive Search

We also implemented an exhaustive search of the entire parameter space to compare its running time to our fast simulated annealing algorithm. The comparison of our simulated annealing algorithm and exhaustive search drastically demonstrates the advantage of simulated annealing. We used image Noentry2 which has 112×77 pixels. The search space had about 6.8×10^7 sets of parameters. It took 15 seconds to recognize the sign using our simulated annealing algorithm. In contrast, exhaustive search found the sign after more than 10 hours of computation time.

Figure 7 illustrates how fast our simulated annealing algorithm recognizes a sign in a scene image.

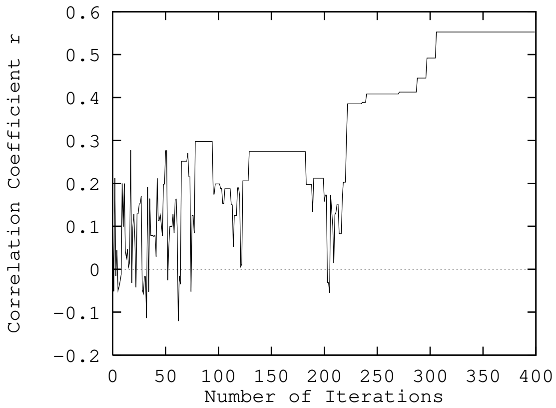

Figure 7: A typical run of our simulated annealing algorithm. The sign is found after about 300 iterations (ca. 18 s).

7 Avoiding False Matches

The error in the sampled coefficient of correlation r increases if the number of pixels p_T in the image window considered decreases. For large samples of p_T pixels the error of r can be expressed as the mean squared error (MSE)

$$E[(r-\rho)^2] = \frac{1-\rho^2}{\sqrt{p_T}}$$

(see Figure 8 and Weatherburn [Wea62]). As Weatherburn points out, the sampling distribution of r is never even approximately normal. The probability curve is very skewed in the neighborhood of $\rho = \pm 1$, even for large samples.

The *normalized auto-correlation* of model image $M(x, y)$ is

$$R(\tau_x, \tau_y) = \frac{\sum_x \sum_y M(x,y) M(x-\tau_x, y-\tau_y)}{\sum_x \sum_y (M(x,y))^2}.$$

The faster the auto-correlation falls off, the higher the resolution of the model image. Examples of auto-correlation images are shown in Figure 9. The resolu-

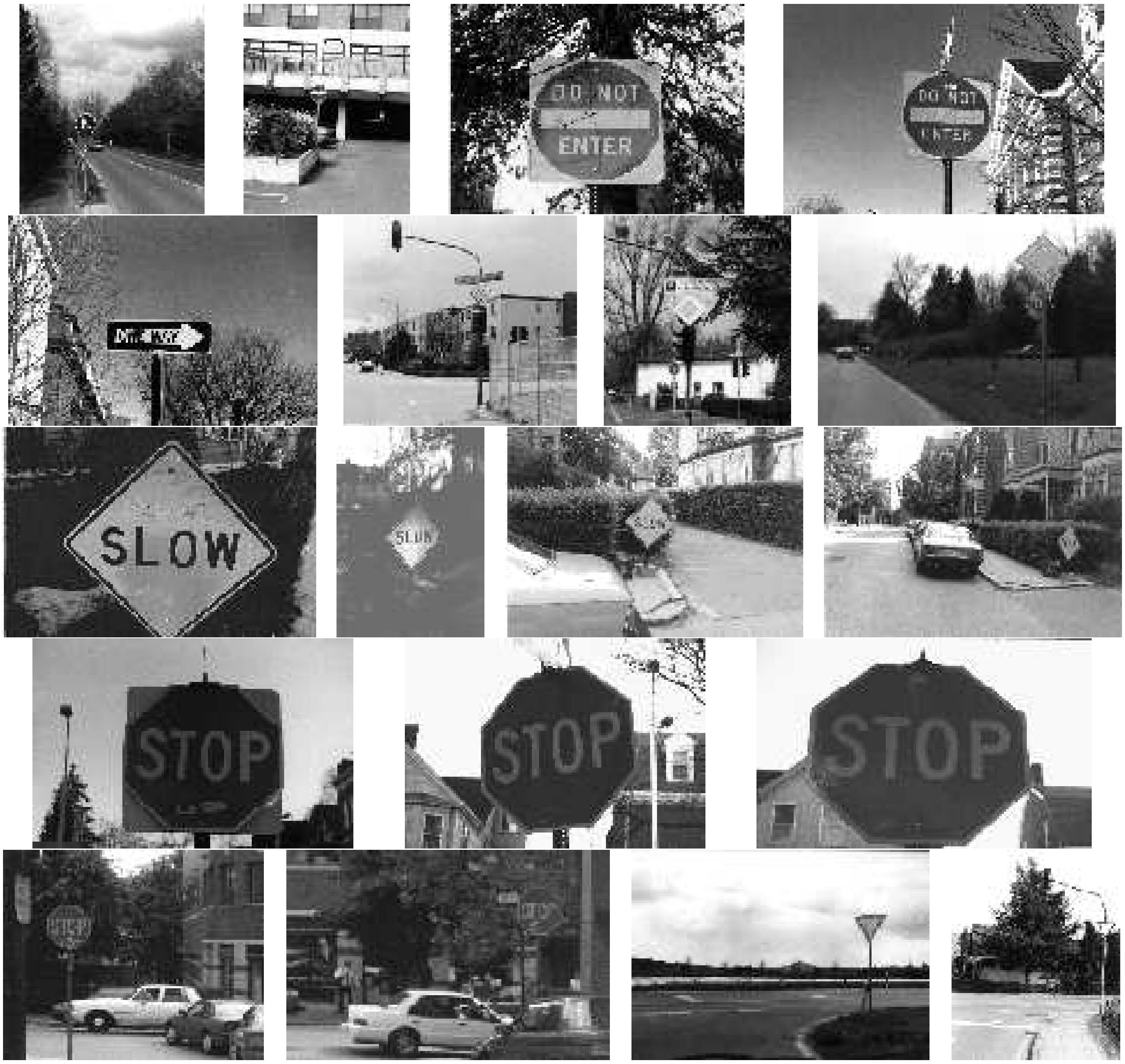

Figure 5: Scene images used in recognition experiments. The images are named by the sign in the scene and a number if the same sign is in more than one scene image. Reading left to right, the images are: Footpath, E-no-entry, No-entry 1 & 2, One-way, Priority 1, 2, & 3, Slow 1, 2, 3, & 4, Stop 1, 2, 3, 4, & 5, and Yield 1 & 2.

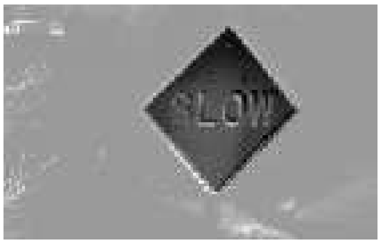

Image 1

Image 2

Image 3

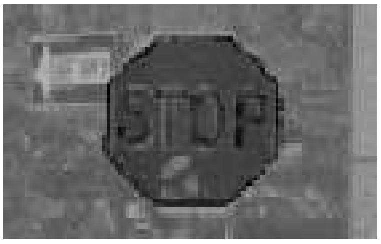

Image 4

Image 5

Image 6

Figure 6: False positive matches: Images 1 and 2 show templates constructed from models Slow and Yield overlying the sign in image Slow2 (correlation values 0.56 and 0.58, respectively.) Images 3 and 4 are cropped images of Stop4 and Stop5 illustrating the best match with templates made from the Stop model. For images Stop4 and Stop5, we obtain better correlation values using models E-no-entry and Yield. Cropped versions of image Stop5 illustrating these false positive matches are shown in Images 5 and 6.

TABLE 1
Correlation Values for Recognition Task

	Models							
Images	Footpath	E-no-entry	No-entry	One-way	Priority	Slow	Stop	Yield
Footpath	**0.77**	0.59	0.38	0.37	0.46	0.29	0.35	0.62
E-no-entry	0.49	**0.73**	0.39	0.43	0.46	0.26	0.38	0.62
No-entry1	0.22	0.21	**0.67**	0.31	0.24	0.18	0.17	0.40
No-entry2	0.29	0.18	**0.84**	0.37	0.14	0.26	0.23	0.35
One-way	0.37	0.55	0.24	**0.70**	0.40	0.38	0.31	0.58
Priority1	0.36	0.49	0.34	0.35	**0.58**	0.32	0.30	0.44
Priority2	0.46	0.54	0.40	0.45	**0.66**	0.29	0.32	0.31
Priority3	0.37	0.57	0.40	0.39	**0.62**	0.34	0.37	0.56
Slow1	0.25	0.29	0.25	0.25	0.45	**0.74**	0.15	0.38
Slow2	0.38	0.48	0.39	0.39	0.32	0.56 2nd	0.21	**0.58**
Slow3	0.39	0.58	0.41	0.38	0.40	**0.62**	0.30	0.59
Stop1	0.41	0.47	0.42	0.30	0.22	0.25	**0.69**	0.58
Stop2	0.23	0.16	0.27	0.25	0.18	0.11	**0.38**	0.30
Stop3	0.26	0.20	0.33	0.19	0.13	0.00	**0.34**	0.19
Stop4	0.42	**0.73**	0.46	0.50	0.43	0.32	0.56 3rd	0.66
Stop5	0.43	**0.73**	0.44	0.48	0.29	0.31	0.51 3rd	0.65
Yield1	0.45	0.75	0.39	0.50	0.53	0.32	0.37	**0.78**
Yield2	0.42	0.73	0.39	0.50	0.43	0.32	0.36	**0.82**

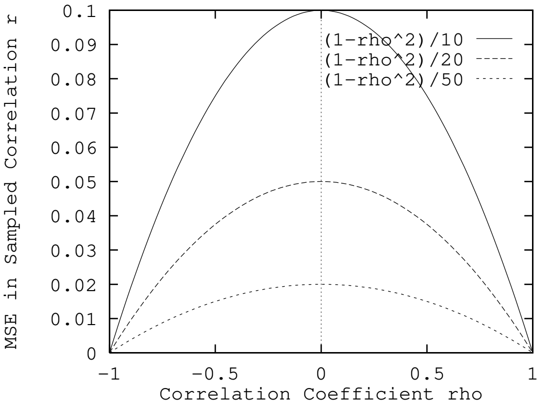

Figure 8: Mean squared error of r for $p_T = 100, 400$ and 2500.

tion of a given model image can be measured with a single number, the *coherence area* $A = \sum_x \sum_y (R(x,y))^2$. Given the coherence area A and the number of pixels n of $M(x,y)$, the number of *coherence cells* is $c = n/A$. The number of coherence cells is equivalent to the number of degrees of freedom of the model image. It can be used as a measure of the information content of the model image.

We examine the information content of each model image to evaluate how useful the model image is for the recognition task. All our model images $M(x,y)$ have the same number of pixels n. Model images with low resolution (little structure) such as the European No-entry and Yield signs, do not have enough information content for robust object recognition. This, and the mean squared error in r for small p_T, are responsible for the false positive matches reported in Table 1. In order to avoid false matches, we need to avoid using such model images with low information content.

The models that contribute to the false positive matches, E-no-entry and Yield, have a coherence area of 313 and 197, respectively. This is much higher than the coherence area for models with more reliable matching results. For example, the Footpath and Stop signs' auto-correlation falls off much faster; their coherence areas are 148 and 56, respectively. The number of coherence cells in E-no-entry is 297 and in Yield 473, but in Footpath it is 628 and in Stop, even 1641.

Thus, the number of coherence cells is a quantitative measure for determining if a model has enough information content to be useful as a template. Most of the models we use have a large enough number of coherence cells for robust detection, but subsequent downsampling in generation of templates may corrupt this.

8 Results on Noisy Images

Gaussian noise is added to the brightness values of some of the scene images to examine the robustness of our algorithm. The algorithm is able to find the sign even in strongly degraded pictures. The signal-to-noise ratio (SNR) of a noisy image is defined as 10 log of the variance of the noisy image over the variance of the noise.

Several noisy images are obtained by corrupting image Slow3 by zero-mean Gaussian noise with various signal-to-noise ratios. Our results for image Slow3 are summarized in Figure 10. Note that the correlation increases as the signal-to-noise ratio increases.

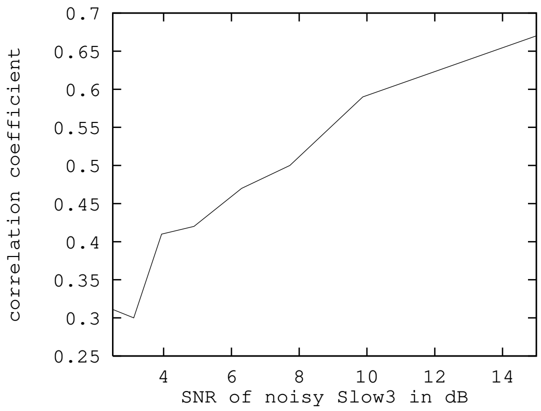

Figure 10: Correlation coefficient for sign recognition in noisy versions of image Slow3.

Figure 11 shows images Slow3 and Slow4 corrupted by Gaussian noise with zero mean and SNR 3 dB and 5 dB, respectively. Matches for pictures with much lower SNR are possible for templates with much larger number of pixels and information content than those presented. (In radar and sonar, signals with negative SNR are commonly extracted given sufficient information content.)

9 Conclusions

Our method has been shown to efficiently recognize objects in complicated landscapes in the presence of noise. To our knowledge, our work is the first to apply fast simulated annealing to object recognition. Our results show that it makes the parameter search of object recognition feasible.

We strongly advocate the use of template matching in recognition tasks and provide quantitative techniques to analyze its limits. We show how to measure the information content of templates as a way to make the recognition algorithm robust.

For the application of traffic signs, we have shown that the search space can be successfully reduced by using a three parameter transformation from model image to template. This method is well suited for recognition tasks that involve objects with scale and shape variations. The method is so efficient that templates can be constructed on-line during the search.

For future work, severe illumination variations within the object and occlusion problems can be addressed. Other applications of our method, for example in medical computer vision and in face recognition, are being investigated. A recent paper by Brunelli and Poggio [BP93] reports successful face recognition using template matching. The authors normalize their test images by fixing the direction of the eye-to-eye axis and the interocular distance. The location of the masks for eye, nose, mouth, and face templates are also fixed. We believe that we can

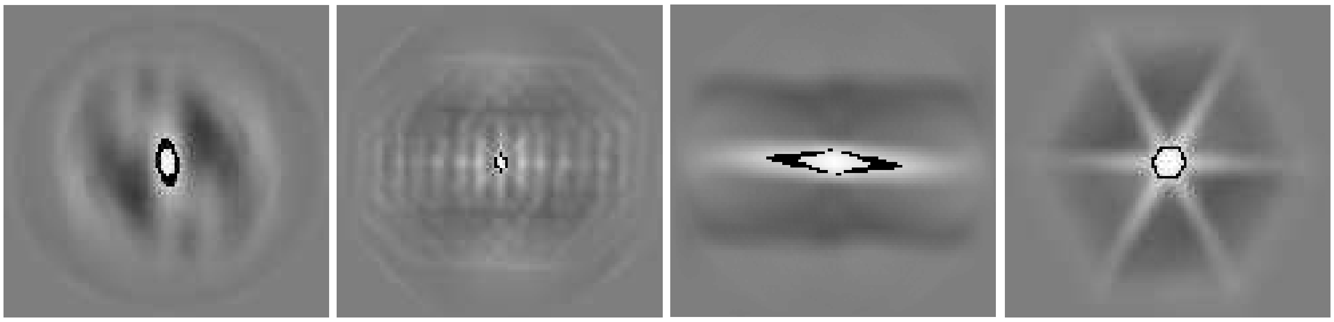

Figure 9: Auto-correlation of model images Footpath, Stop, E-no-entry, and Yield. To illustrate how fast the auto-correlation falls off, the e-folding lengths, i.e., pixels (x, y) with $R(x, y) \approx 1/e$, are shown on a dark contour.

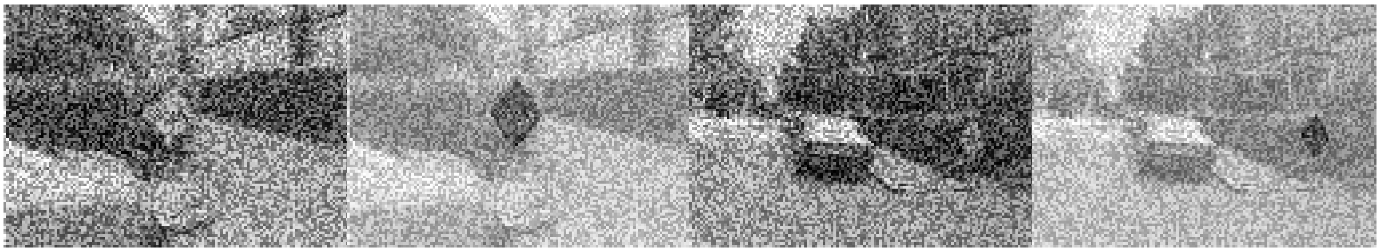

Figure 11: The first and third images are images Slow3 and Slow4 degraded by Gaussian noise with zero mean and SNR 3 dB and 5 dB, respectively. The second and fourth images illustrate that the object is recognized where the templates computed are shown overlying the recognized sign in the scene. (These images are shown brighter so that the overlying template can be illustrated better.)

generalize Brunelli and Poggio's application to recognize faces in images that are not normalized but contain more general scenes with varied backgrounds.

Acknowledgements

We would like to thank Wilfried Betke for taking some the pictures, and Marney Smyth, Michelle Hsu, Ou-Dan Peng, and Acee Agoyo for their help preparing the document.

References

[AF86] Nicholas Ayache and Olivier D. Faugeras. HYPER: A new approach for the recognition and positioning of two-dimensional objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(1):44–54, January 1986.

[BB82] Dana H. Ballard and Christopher M. Brown. *Computer Vision*. Prentice-Hall, 1982.

[BG94] Margrit Betke and Leonid Gurvits. Mobile robot localization using landmarks. *IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, September 1994. Also published as Technical Report SCR-94-TR-474, Siemens Corporate Research.

[BKM93] Arthur B. Baggeroer, William A. Kuperman, and Peter N. Mikhalevsky. An overview of matched field methods in ocean acoustics. *IEEE Journal of Oceanic Engineering*, 18(4):401–424, 1993.

[BP93] Roberto Brunelli and Tomaso Poggio. Face recognition: Features versus templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):1042–1052, October 1993.

[Bra90] David J. Braunegg. MARVEL: a system for recognizing world locations with stereo vision. Technical Report 1229, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, May 1990.

[CBK+93] Michael D. Collins, Jonathan M. Berkson, William A. Kuperman, Nicholas C. Makris, and John S. Perkins. Applications of optimal time-domain beamforming. *Journal of the Acoustical Society of America*, 93(4):1851–1865, April 1993.

[Ett88] Gil J. Ettinger. Large hierarchical object recognition using libraries of parameterized model sub-parts. In *IEEE Proceedings of Computer Vision and Pattern Recognition*, pages 32–41, June 1988.

[FR91] Noah S. Friedland and Azriel Rosenfeld. Lobed object delineation using a multipolar representation. Technical Report CS-TR-2779, Center of Automation Research, University of Maryland, October 1991.

[GG84] Stuart Geman and Donald Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741, November 1984.

[GGGD90] Donald Geman, Stuart Geman, Christine Graffigne, and Ping Dong. Boundary detection by constrained optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):609–628, July 1990.

[Gri88] W. Eric L. Grimson. On the recognition of parameterized 2D objects. *International Journal on Computer Vision*, 3:353–372, 1988.

[Gri90] W. Eric L. Grimson. *Object Recognition by Computer: The Role of Geometric Constraints.* MIT Press, 1990.

[Joh82] J. H. Johnson. The application of spectral estimation methods to bearing estimation problems. *Proceedings of the IEEE*, 70(9):1018–1028, 1982.

[KCPD90] William A. Kuperman, Michael D. Collins, John S. Perkins, and N.R. Davis. Optimal time-domain beamforming with simulated annealing including application of a priori information. *Journal of the Acoustical Society of America*, 88(4):1802–1810, October 1990.

[KGV83] S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi. Optimization by simulated annealing. *Science*, pages 671–680, May 1983.

[LC88] Z.-Q. Liu and Terry M. Caelli. Multiobject pattern recognition and detection in noisy backgrounds using a hierarchical approach. *Computer Vision, Graphics, and Image Processing*, 44:296–306, 1988.

[May94] Franz May. Vision system for safe driving. Presented at the Center for Biological and Computational Learning, MIT, September 1994.

[MR90] Avraham Margalit and Azriel Rosenfeld. Using probabilistic domain knowledge to reduce the expected computational cost of template matching. *Computer Vision, Graphics, and Image Processing*, 51:219–234, 1990.

[MRR+53] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equations of state calculations by fast computing machines. *J. Chem. Physics*, C–21, 1953.

[NR72] Roger N. Nagel and Azriel Rosenfeld. Ordered search techniques in templates matching. *Proceedings of the IEEE*, 60(2):242–244, 1972.

[Opp78] A. V. Oppenheim, editor. *Applications of Digital Signal Processing.* Englewood Cliffs: Prentice Hall, 1978.

[SH87] H. Szu and R. Hartley. Fast simulated annealing. *Physics Letters A*, 122(3–4):157–162, June 1987.

[Wea62] C. E. Weatherburn. *A First Course in Mathematial Statistics.* Cambridge University Press, 1962.

[Yar85] Leonid P. Yaroslavsky. *Digital Picture Processing.* Springer-Verlag Berlin, 1985.

[ZRJ94] Yong-Jian Zheng, Werner Ritter, and Reinhard Janssen. An adaptive system for traffic sign recognition. In *Proceedings of the Intelligent Vehicles Symposium*, 1994.