Appeared in Proc. 4th European Conf. on Computer Vision, Lecture Notes in Computer Science 1064, pp556-565, London, 1996

Image Recognition with Occlusions

Tyng-Luh Liu[1] Mike Donahue[2] Davi Geiger[1] Robert Hummel[1]

[1] Courant Institute, New York University, New York NY 10012, USA
[2] IMA, University of Minnesota, Minneapolis MN 55455, USA

Abstract. We study the problem of how to detect "interesting objects" appeared in a given image, I. Our approach is to treat it as a function approximation problem based on an over-redundant basis. Since the basis (a library of image templates) is over-redundant, there are infinitely many ways to decompose I. To select the "best" decomposition we first propose a global optimization procedure that considers a concave cost function derived from a "weighted L^p norm" with $0 < p \leq 1$. This concave cost function selects as few coefficients as possible producing a sparse representation of the image and handle occlusions. However, it contains multiple local minima. We identify all local minima so that a global optimization is possible by visiting all of them. Secondly, because the number of local minima grows exponentially with the number of templates, we investigate a greedy "L^p Matching Pursuit" strategy.

1 Introduction

In the field of signal processing and computer vision an input signal or image is a function f over some subset of $\mathbb{R}$ or $\mathbb{R}^2$. To manipulate and analyze f, it is useful to introduce a linear decomposition into basis elements f_j, i.e., $f = \sum_j c_j f_j$. An example of a well known and useful decomposition of this type is the Fourier series expansion.

We study the object recognition problem via a robust template decomposition approach. Let the image to be recognized be I and the template library be $\mathcal{L}$. The task of image recognition is reduced to a function approximation problem of the form

$$I(x) = \sum_j \sum_i c_{ij} A_i(\tau_j)(x) = \sum_{i,j} c_{ij} T_{ij}(x) \tag{1}$$

where $\tau_j \in \mathcal{L}$, $T_{ij} = A_i(\tau_j)$ denotes an affine transformation applied to the template τ_j, and c_{ij} is the choice of coefficients that "best" decompose the image. Typically the library $\mathcal{L}$ is large, in order to accommodate many possible situations and also consider the possible (affine) transformations. Thus, we have an over-redundant basis leading to infinite many solutions, c_{ij}, to this problem. That is not the case for the Fourier decomposition.

Let us illustrate the problem of function decomposition with over-redundant library. Say our basis consists of sinusoids and functions of the form $1/(k + x)$ $(k \in \mathbb{N})$. Assume that $f(x) = \sin 2x + \frac{4}{(3+x)}$ is our target function (our image). It is clear that only two terms from the prototype library are required to represent $f(x)$. However, one could write $f(x)$ using either sinusoids alone or as combinations of $1/(k + x)$ alone, but either representation would require many

terms. The problem is to formulate a coefficient selection criterion and a method to compute the coefficients that yields compact representations.

1.1 Coefficient selection, concave cost function, and optimization

Our approach [7, 13] is to construct an objective function $F(\mathbf{c})$ that when minimized selects a best representation, $\mathbf{c}^*$, from among all solutions $\mathbf{c}$ that satisfy the constraint $I(x) = \sum_j \sum_i c_{ij} A_i(\tau_j)(x)$. We require

1. **Sparse Representation:** represent (decompose) an image using as few templates as possible in order to have an economical (minimal) representation. Field [9] also argued for sparse representations in the brain.

2. **Occlusions:** allow for partial occlusions, i.e., the cost of fitting a template must take into account that portions of the template may have a "bad match".

3. **Noise:** model noise via "noise templates" accounting for the difference between the template fit and the image. This leads us to search for cost functions that escalate with the magnitude of c_{ij}, but should not dominate the first condition, i.e., the rate of increase in cost as a function of $|c_{ij}|$ should decrease.

The above consideration leads us naturally to adopt concave objective functions. In particular, we will primarily study the objective function

$$F_p(\mathbf{c}) = \sum_{j=1}^{M} \sum_{i=1}^{N} \omega_{ij} |c_{ij}|^p \,, \tag{2}$$

where N is the number of possible (affine) transformations and M is the size of the template library. The scalars ω_{ij}'s are positive, e.g., they may be set to 1 or to the inverse of the template and image variances.

The sparsity of templates suggests $\mathbf{p} = 0$ to count the number of templates (weighted by ω_{ij}). Noise templates should be paid according to how large the "repair" is, i.e., how large the error c_{ij} is. The balance between both processes, sparsity of the templates and noise modeling leads to values of $0 < \mathbf{p} \leq 1$.

The objective function is non-convex, and in fact the optimization problem will generally have multiple local minima, making the optimization more difficult. We will show that it is possible to characterize all local minima and obtain the global one by visiting them. Since the number of local minima grows exponentially with the size of the template library we consider an alternative greedy algorithm. Recently, Chen and Donoho [3, 4] studied the overcomplete signal representation problems with L^1 norm optimization. Their method is based on linear programming, which is efficient, but only applies to the $\mathbf{p} = 1$ case and still leads to a slow algorithm. Coifman and Wickerhauser [5], modeled an entropy like function, $\sum_{i,j} |c_{ij}|^2 log|c_{ij}|^2$ with more constraints on the the coefficients c_{ij} square-sum to 1.

Comparison with principal component analysis/Eigenfaces: Our approach is fundamentally different from the "eigenfaces" approach (PCA approach) [16]. In our case the basis functions are fixed and the adaptation of the method is on choosing the appropriate coefficients (from a redundant basis), a non-linear process. In the PCA approach the choice of basis functions, a linear process, is where the adaptation occurs. PCA works well only when the task function is a simple linear superposition of the basis functions.

1.2 Matching pursuit

Inspired by Mallat and Zhang's work [14] we consider a matching pursuit strategy where, at each stage, the criterion of best selection is based on minimizing an image residue. In regression statistics, this decomposition method is known as *Projection Pursuit Regression* , a non-parametric method that is concerned with "interesting" projections of high dimensional data (see Friedman and Stuetzle [10], Huber [11]). Recently, Bergeaud and Mallat [2] used the (L^2) matching pursuit with a redundant family of Gabor oriented wavelets to approximate images and produce compact decompositions for the main features of images.

The original matching pursuit is based on the standard L^2 (Hilbert space) method. We propose an L^p matching pursuit with $0 < \mathbf{p} \leq 1$, to improve the robustness. With $0 < \mathbf{p} \leq 1$, we lose the structure of inner product but the notion of a template "closest" to the image is recaptured via the cost function.

2 Template Library and Image Coordinates

We must first establish a well-defined over-redundant library of templates containing many non-canonical templates as well as one canonical template. A canonical template is a trivial template with zero gray-level value pixels everywhere except one pixel at the extreme left and top corner that its gray-level value is 1. Moreover, we will assume we can apply a set of affine transformations to each template, indeed we will restrict ourselves to translations. Clearly, this single canonical template plus a set of all translations form a basis for the image space.

Coordinate transformations: Suppose we have now created a template library $\mathcal{L} = \{\tau_j : j = 1...M\}$ for some application, where we will use $\epsilon_1 \equiv \tau_1$ to represent the canonical template. Let the image to be recognized be I of dimension N and each template τ_j be of dimension N_T (we assume that both N and N_T are perfect square numbers). Furthermore, let $P = \{p_1, p_2, \cdots, p_N\}$ and $Q = \{q_1, q_2, \cdots, q_{N_T}\}$ be the pixel sets of I and any τ_j, respectively. (We order the pixels from top to bottom and left to right.) Let the translation $A_i(\tau_j)$ indicate that the first template pixel q_1 is positioned at the *i-th* pixel $p_i \in P$. The mapping formula for A_i is such that $q_r \mapsto p_k = p_{k(r,i)}$ where [3] $k = i + (\lfloor \frac{r-1}{\sqrt{N_T}} \rfloor \times N) + (r - 1 - \lfloor \frac{r-1}{\sqrt{N_T}} \rfloor \times \sqrt{N_T})$. Denote $T_{ij} = A_i(\tau_j)$ and $e_{i1} = T_{i1} = A_i(\epsilon_1)$ [4], then we have $T_{ij}(p_k) = \tau_j(q_r)$. Using these notations, one can write the decomposition equation (1) as

[3] The expression $\lfloor x \rfloor$ denotes the greatest integer less than or equal to x.

[4] Note that $e_{i1}(p_j) = e_i(p_j) = \delta_{ij}$, where $\delta_{ij} = 1$ for $i = j$ and $\delta_{ij} = 0$ otherwise.

$$I(p_k) = \sum_{i=1}^{N} c_{i1} e_{i1}(p_k) + \sum_{j=2}^{M} \sum_{i=1}^{N} c_{ij} T_{ij}(p_k) = \sum_{\lambda=1}^{N} c_\lambda e_\lambda(p_k) + \sum_{\lambda=N+1}^{M.N} c_\lambda T_\lambda(p_k) \quad (3)$$

where $\lambda = \lambda(i,j) = (j-1) \times N + i$. We may write $I[k]$, $e_\lambda[k]$ and $T_\lambda[k]$ instead of $I(p_k)$, $e_\lambda(p_k)$ and $T_\lambda(p_k)$, respectively, for simplification.

3 Optimization Problem and Solution

Equation (3) can be written in matrix notation as $\mathrm{T}\mathbf{c} = \mathrm{I}$ where

$$\mathrm{T} = \begin{pmatrix} e_1[1] & \cdots & e_N[1] & T_{N+1}[1] & \cdots & T_{MN}[1] \\ e_1[2] & \cdots & e_N[2] & T_{N+1}[2] & \cdots & T_{MN}[2] \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ e_1[N] & \cdots & e_N[N] & T_{N+1}[N] & \cdots & T_{MN}[N] \end{pmatrix},$$

$$\mathbf{c} = (c_1, c_2, \ldots, c_{MN})^t \qquad \text{and} \qquad \mathrm{I} = (I[1], I[2], \ldots, I[N])^t .$$

Note that if the prototype library forms a basis (linearly independent), then $M = 1$, and there is no freedom in choosing the coefficients (c_λ); the coefficients are uniquely determined by the constraint. If there are linear dependencies in the prototype library, then $M > 1$, the prototype library over-spans, and the set of all solutions (c_λ) to the constraint forms an $(M-1)\,N$ dimensional affine subspace in the $M.N$-dimensional coefficient space. Let S denote this solution space, i.e., $\dim(S) = (M-1)\,N$. Using the above matrix notations, our optimization problem can be formulated as:

$$\min_{\mathbf{c}} F_p(\mathbf{c}) = \min_{\mathbf{c}} \sum_{\lambda=1}^{MN} \omega_\lambda |c_\lambda|^p \qquad \text{subject to the constraint } \mathrm{T}\mathbf{c} = \mathrm{I} \qquad (4)$$

where $\mathrm{T} \in \mathbb{R}^{N \times M.N}$, $\mathbf{c} \in \mathbb{R}^{M.N}$, $\mathrm{I} \in \mathbb{R}^N$, $M > 1$. The next result is shown in [7, 13], or previously stated in [8].

Proposition 1 *All the local minima of L^p-cost function in (4) occur at the vertices of a polytope. This polytope is constructed from the intersection of the affine subspace S and a cube defined by the origin and bounded in each axis by d_λ. d_λ can be as large as $(F_p(\mathbf{c}_0)/\omega_\lambda)^{1/p}$, where $\mathbf{c}_0$ is any solution to $\mathrm{T}\mathbf{c} = \mathrm{I}$.*

4 One Template Matching and Simulations

If we want to find a specific face in an image, then it suffices to use only one face-template. In these cases the non-canonical template represents a key feature and the canonical templates e_λ represents non-interest elements , e.g., noise. Let us assume that this particular template be τ_2 of size N_T ($\tau_1 \equiv \epsilon_1$) and A_i

be the translation, that is, $A_i(\tau_2) = T_{i2} = T_{N+i}$. This says that we look for a decomposition of the form:

$$I(x) = c_{N+i}T_{N+i}(x) + \sum_{\lambda=1}^{N} c_\lambda e_\lambda(x)\,. \tag{5}$$

It is clear that $c_\lambda = I[\lambda]$ if pixel p_λ is not covered by T_{N+i}. So, the equation (1) can be restricted to the region where T_{N+i} is located.

$$\begin{pmatrix} e_{A_i(1)}[A_i(1)] & \cdots & e_{A_i(N_T)}[A_i(1)] & T_{N+i}[A_i(1)] \\ e_{A_i(1)}[A_i(2)] & \cdots & e_{A_i(N_T)}[A_i(2)] & T_{N+i}[A_i(2)] \\ \vdots & \ddots & \vdots & \vdots \\ e_{A_i(1)}[A_i(N_T)] & \cdots & e_{A_i(N_T)}[A_i(N_T)] & T_{N+i}[A_i(N_T)] \end{pmatrix} \begin{pmatrix} c_{A_i(1)} \\ \vdots \\ c_{A_i(N_T)} \\ c_{N+i} \end{pmatrix} = \begin{pmatrix} I[A_i(1)] \\ \vdots \\ I[A_i(N_T)] \end{pmatrix}$$

where $e_i[j] = e_i(p_j) = \delta_{ij}$. Recall that $T_{N+i}[A_i(r)] = \tau_2[r]$ (and $A_i(1) = i$). We can also assume that $\tau_2[r] \neq 0$ for $r = 1, \ldots, N_T$, since otherwise we can redefine either τ_2 or the pixel ordering to get a smaller value for N_T.

It follows from Proposition 1 that the local minima of $F_p(\mathbf{c})$ can be found by setting c_{N+i}, $c_{A_i(1)}$, $\ldots$, $c_{A_i(N_T)}$ to zero one at a time. If we set $c_{N+i} = 0$ then we get $c_\lambda = I[\lambda]$ for all λ. This is the "pure noise" solution. The first nontrivial (template using) solution sets $c_{A_i(1)} = 0$. This forces the template coefficient $c_{N+i} = I[A_i(1)]/\tau_2[1]$, from which it follows that $c_{A_i(r)} = I[A_i(r)] - c_{N+i}\tau_2[r]$, for $r = 2, \ldots, N_T$. The solution determined by setting $c_{A_i(r)} = 0$ $(2 \leq r \leq N_T)$ can be calculated in an analogous fashion.

The optimal cost of the match of the template in the (translation) position i is the smallest of the values of $F_p(\mathbf{c})$ across all $N_T + 1$ solutions $(\mathbf{c})$. One performs a similar analysis for all template translations, and finds the position which generated the smallest match cost. Note that in the case of one template matching, the L^p-norm decomposition problem is actually the same as p-norm minimization.

4.1 Simulations

We have designed a sequence of experiments focused on the effects of noise and occlusions to demonstrate both the weighted and unweighted (all ω_λ's are set to 1) L^p decomposition methods are superior to the conventional correlation techniques. The weights used in the weighted scheme are defined as $\omega_{\lambda(i,j)} = 1/([\sum_{k=1}^{N_T} |\tau_j[k]|^p][\sum_{k=1}^{N_T} |I[A_i(k)]|^p])$, for $0 < \mathbf{p} \leq 1$.

The experiments consist of numerous trials on random images with fixed occlusion size and fixed noise variance. The latter determines the signal-to-noise ratio (SNR) for the experiment, defined here as the ratio of the standard deviation of the image to the standard deviation of the noise.

Each trial has four components: an image, a template, an occlusion, and noise. The image is 64 pixels wide by 64 pixels high, randomly generated using an uncorrelated uniform distribution across the range $(-256, 256)$. The template is a 4 pixel by 4 pixel subimage of the image. After selecting the template, a portion of the image from which the template is drawn is "occluded" by redrawing from

the same distribution that formed the image, i.e., from an uncorrelated uniform distribution with range $(-256, 256)$. (Occlusion sizes range from 0–14 pixels, from a total subimage size of 16 pixels.) Finally, noise is added to the (occluded) image, drawn from an uncorrelated Gaussian mean-zero random variable.

Translates of the template are compared against the noisy, occluded image, using both weighted and unweighted L^p-norm decomposition method. (Because both the template and the image are drawn from zero-mean random variables, there is little difference between 2-norm error minimization and standard correlation.) For each method the translation position yielding the best score is compared with the position of the original subimage from which the template was formed. If the two agree then the match is considered successful, otherwise the match fails for the trial in question.

5 Multiple Templates and Matching Pursuit

In this section, we proceed to elucidate the matching pursuit method for the case of multiple templates. The basic idea is to devise a greedy iterative method where at each stage only one template is selected and thus, we can rely on the previous section result. In this section we will also consider, for comparison, a cost function based on the LTS (Least Trimmed Squares, Rousseuw 1983, 1984, [15]).

5.1 Review

We briefly review the (L^2) matching pursuit below. Suppose it is given a signal f, and a library of functions $D = \{g_\gamma\}_{\gamma \in \Gamma}$ where Γ is a set of index tuples and D represents a large, over-redundant family of functions. A "best" matching library element to the residual signal structures at each stage is decided by successive approximations of the residual signal with orthogonal projections on elements in the library. That is, say at stage n, for any element $g_\gamma \in D$, we consider

$$R^{n-1}f = < R^{n-1}f, g_\gamma > g_\gamma + R^n f \tag{6}$$

where $R^n f$ is the *n-th* residue after approximating $R^{n-1}f$ in the direction of g_γ (assume that the initial residue is the function f, i.e. $R^0 f = f$). The matching pursuit strategy is to find g_{γ^*} that minimizes $||R^n f||$ (or the g_{γ^*} closest to $R^{n-1}f$), i.e. $||R^{n-1}f - < R^{n-1}f, g_{\gamma^*} > g_{\gamma^*}||_{L^2} = \min_{\gamma \in \Gamma} ||R^{n-1}f - < R^{n-1}f, g_\gamma > g_\gamma||_{L^2}$.

5.2 Our approach

Assume that $R^0 I = I$, the input image. Then, at stage n, if a transformed template $T_\lambda (= T_{ij} = A_i(\tau_j))$ and coefficient c_λ are chosen, the *n-th* residual image can be updated as follows:

$$R^n I(p_k) = R^{n-1} I(p_k) - c_\lambda T_\lambda(p_k) \qquad \text{for } k = 1 \ldots N\,. \tag{7}$$

Note that T_λ is only of dimension N_T and we assume that $T_\lambda(p_k) = 0$ if p_k is not covered by T_λ. From (7), $R^n I$ can be derived by "projecting" $R^{n-1}I$ in the direction of T_λ. At each stage, we recover a best matching by minimizing $\omega_\lambda ||R^n I||_{L^p}$ where ω_λ is defined similarly to the case of one-template matching.

Appeared in Proc. 4th European Conf. on Computer Vision, Lecture Notes in Computer Science 1064, pp556-565, London, 1996

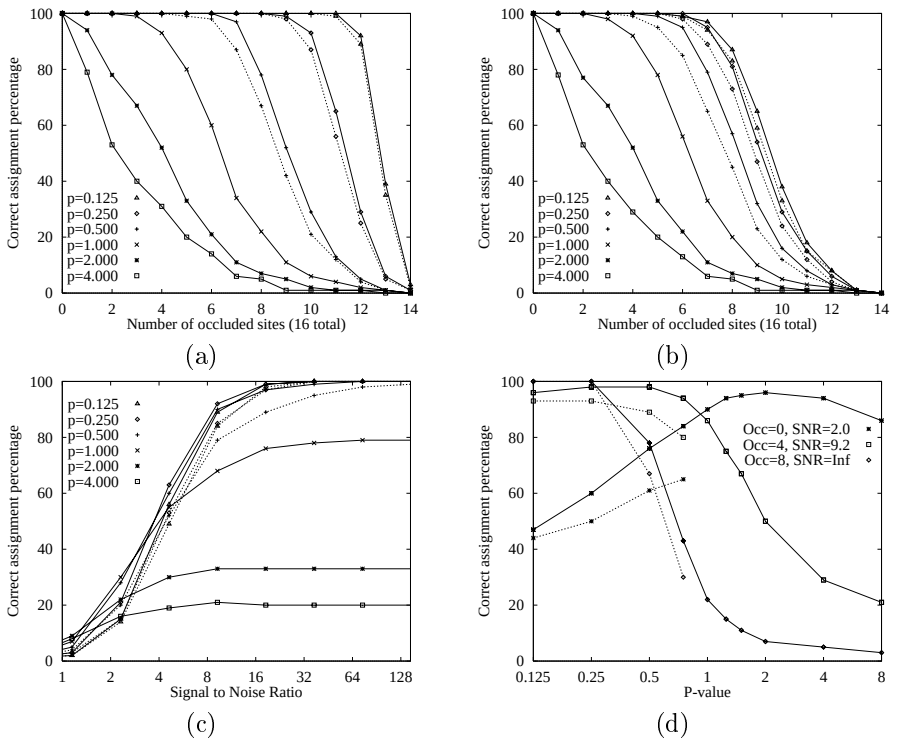


Fig. 1 (a)Experimental matching accuracy as a function of occlusion size with no noise. The solid curves correspond to our proposed weighted L^p decomposition, dashed to the unweighted L^p decomposition. Note that smaller values of $\mathbf{p}$ outperform larger values providing nearly 100% correct results with $\mathbf{p} = 0.125$ for occlusions as large as 11 (out of 16) pixels. Also given a $\mathbf{p}$ value the weights (normalization) can help improve the results. (b) Experimental matching accuracy as a function of occlusion size at a Signal to noise Ratio (SNR) of 37. The solid curves correspond to our proposed method, dashed to the unweighted L^p method. Here we note that $\mathbf{p} = 0.125$ still performs very well, although good results can not be obtained if the occlusion is larger than half the template size. Notice that the results using larger values of $\mathbf{p}$ are less affected by noise, especially those with $\mathbf{p} > 1$. (c) Experimental matching accuracy as a function of noise level at a fixed occlusion size of 5 (out of 16) pixels. Note again that larger values of $\mathbf{p}$ produce results which are less sensitive to noise. For example, the results for $\mathbf{p} = 0.125$, which are best for large SNR, are poorest for SNR of less than about 3. The solid (dashed) curves correspond to weighted (not weighted) L^p method. (d) Experimental matching accuracy as a function of $\mathbf{p}$, at various noise levels and occlusion sizes. For an occlusion size of 4/16 and a SNR of 9.2, the best $\mathbf{p}$ value for our weighted L^p method is somewhere between 0.25 and 0.5. The solid (dashed) curves correspond to weighted (not weighted) L^p method.

(a) (b) (c) (d) I_1 (e) R_1 (f) R_2

Fig. 2 (a), (b), (c) are synthetic template type 1, type 2 and type 3, respectively. (d) Test image I_1 with noise added and occlusion (e) Result of the decomposition for the L^p with $\mathbf{p} = 0.25$, and also for the LTS with $\alpha = 0.51$. (α is the robust constant of LTS.) (f) Results once the breakdown limits are reached, and occluded templates are not recognized. For example, L^p, with $\mathbf{p} = 0.75$ and LTS with $\alpha = 0.75$.

5.3 Matching pursuit simulations

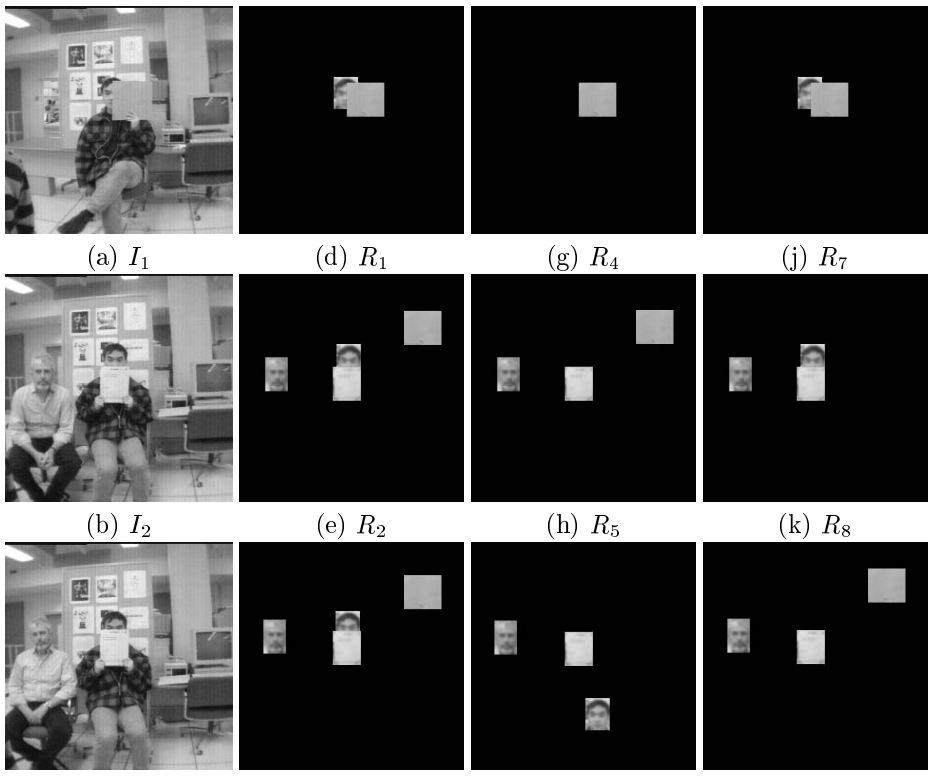
We first work with synthetic data and then with real images.

Synthetically Randomized Images : Let's begin with a simple experiment to test our template matching algorithm for a synthetic example. In this experiment, the template library $\mathcal{L}$ consists of three different types (or shapes) of templates ((a), (b), (c) in Figure 2). There are 40 templates for each type so that $\mathcal{L}$ includes 120 non-canonical templates and one canonical template ϵ_1. Each of the non-canonical template is a synthetically randomized image with gray-level values between $(0, 200)$ generating from a random number generator. To construct a test image I_1 (as in Figure 2-(d)), we first select one non-canonical template randomly from each template type in $\mathcal{L}$ to form the base (exact) image then add noise and an occluded square derived from uniform distribution in $(0, 10)$ and $(245, 255)$, respectively. The threshold values used in simulation vary with respect to the value of $\mathbf{p}$ for L^p matching pursuit and α for LTS matching pursuit. We see that both methods can handle occlusions (e.g. see Figure 2-(e) R_1). Our experiment results suggest for $\mathbf{p} \in (0.25, 0.75)$ and $\alpha \in (0,51, 0.75)$, both the L^p and LTS methods are rather robust. But, as shown in Figure 2-(f) R_2, both methods failed to recognize the occluded object for $\mathbf{p} \geq 0.75$ and for $\alpha \geq 0.75$.

Face Recognition : A small library of face templates has been established (see Figure 3 (a)-(f)). The dimension of all the six templates is 64×64. Numerous experiments have been carried out to test our algorithm. To illustrate, consider the three real images, I_1 - I_3, in Figure 4 (a)-(c). We obtained decomposition results R_1, R_2 and R_3 shown in Figure 4, for $\mathbf{p} = 0.25$. (Similar results are derived for $\mathbf{p} = 0.50$ and 0.75.) When $\mathbf{p} = 2$, it is indeed the L^2 matching pursuit method and the recognition results are R_4, R_5 and R_6. Our proposed L^p matching pursuit has the robustness advantage over the L^2 one. In case that an image contains objects with large occlusions (like I_3), the LTS may fail to recognize them as shown in 4-(l). In addition, the L^p is more efficient than LTS regarding to the computation complexity.

(a) (b) (c) (d) (e) (f)

Fig. 3 (a) - (f) are the face and book templates used in the face recognition simulation.



(a) I_1 (d) R_1 (g) R_4 (j) R_7

(b) I_2 (e) R_2 (h) R_5 (k) R_8

(c) I_3 (f) R_3 (i) R_6 (l) R_9

Fig. 4 (a)-(c) The test images, where some templates are present with **small** distortions (scale and viewing angle), noise and occlusions. (d)-(f) Image decomposition for L^p matching pursuit with $\mathbf{p} = 0.25$ (similar results are obtained for p up to 0.75). (g) - (i) Image decomposition for $\mathbf{p} = 2.0$ and recognition is destroyed (this is equivalent to use correlations methods, like in the L^2 matching pursuit). Note that false recognition occurs in (i). (j)-(l) Image decomposition with LTS.

Appeared in Proc. 4th European Conf. on Computer Vision, Lecture Notes in Computer Science 1064, pp556-565, London, 1996

References

1. J. Ben-Arie and K. R. Rao, *"On the Recognition of Occluded Shapes and Generic Faces Using Multiple-Template Expansion Matching"*, Proceedings IEEE International Conference on Pattern Recognition, New York City, 1993.
2. F. Bergeaud and S. Mallat, *"Matching Pursuit of Images"*, SPIE, Orlando, 1995.
3. S. Chen and D. Donoho, *"Atomic Decomposition by Basis Pursuit"*, Technical Report, Stanford University, May, 1995.
4. S. Chen and D. Donoho, *"Basis Pursuit"*, TR, Stanford University, Nov. 1994.
5. R. Coifman and V. Wickerhauser, *"Entropy-based Algorithms for Best Basis Selection"*, IEEE Transactions on Information Theory, vol. 38, no. 2, 1992.
6. T. H. Cormen, C. E. Leiserson and R. L. Rivest, *Introduction to Algorithms*, McGraw-Hill, 1990.
7. M. J. Donahue and D. Geiger, *"Template Matching and Function Decomposition Using Non-Minimal Spanning Sets"*, Technical Report, Siemens, 1993.
8. H. Ekblom, *"L_p-methods for Robust Regression"*, BIT 14, p.22-32, 1973.
9. D. Field, *"What Is the Goal of Sensory Coding"*, Neural Comp. 6, p.559-601, 1994.
10. J. H. Friedman and W. Stuetzle, *"Projection Pursuit Regression"*, Journal of the American Statistical Association, vol. 76, p.817-823, 1981.
11. P. J. Huber, *"Projection Pursuit"*, The Ann. of Stat., vol. 13, No.2, p.435-475, 1985.
12. P. J. Huber, *Robust Statistics*, John Wiley & Sons, New York, 1981.
13. T. Liu, M. Donahue, D. Geiger and R. Hummel, *"Sparse Representations for Image Decomposition"*, Technical Report, CS, Courant Institute, NYU, 1996.
14. S. Mallat and Z. Zhang, *"Matching Pursuit with Time-Frequency Dictionaries"*, IEEE Trans. on Signal Processing, Dec. 1993.
15. P. J. Rousseeuw and A. Leroy, *Robust Regression and Outlier Detection*, John Wiley, New York, 1987.
16. M. Turk and A. Pentland, *"Eigenfaces for Recognition"*, J. of Cognitive Neuroscience, vol. 3, p.71-86, 1991.
17. B. Uhrin, *"An Elementary Constructive Approach to Discrete Linear l_p-approximation, $0 < p \leq +\infty$"*, Colloquia Mathematica Societatis János Bolyai, 58. Approximation Theory, Kecskemét, 1990.