

Model-driven Selection using Texture

Tanveer Fathima Syeda-Mahmood

Artificial Intelligence Laboratory, M.I.T. Cambridge, MA 02139.¹

Abstract

In this paper we explore the use of texture or pattern information on a 3D object as a cue to isolate regions in an image that are likely to come from the object. We develop a representation of texture based on the linear prediction (LP) spectrum that allows the recognition of the model texture under changes in orientation and occlusions. The candidate matching image regions are obtained without detailed segmentation by a technique called overlapping window analysis. This analysis, under some conditions, guarantees the existence of a window spanning only the model texture regardless of its position and orientation which is sufficient for the recognition of the model texture using the LP spectrum representation. Finally, we evaluate the utility of texture-based selection in combination with other cues such as color in the context of reducing the search involved in recognition.

1. Introduction

A key problem in object recognition is selection, namely, the problem of isolating regions in an image that are likely to come from a single object. It has been shown that such region isolation can considerably reduce the search involved in the matching stage of object recognition [3]. However, the lack of knowledge of illumination conditions and surface geometries of objects in a scene, and the problems of occlusion, shadowing, specularities, and interreflections in the image make it difficult to isolate regions that are likely to come from a single object.

We have been involved in developing an approach to selection using the paradigm of visual attentional selection. Towards this end, an implementation attentional selection has been developed that combines information from a variety of cues such as color and texture to perform selection [8, 10]. In this paper we report on the use of one of the attentional cues, namely, texture to perform selection in a model-driven way, i.e by exploiting a model texture description.

Searching for an instance of a model object texture in an image is a difficult problem as illumination or pose changes, and occlusions can cause the model texture to appear different. In addition, it may not be possible to precisely locate the model texture (i.e. segment it) in the image. Thus a texture-based selection mechanism must be able to account for illumination and pose changes, occlusions, and imprecise localization, and in this sense, becomes a different problem from texture classification [1, 4, 6]. Nevertheless, existing texture analysis methods can be relevant for texture-based selection, such as for example, the Fourier domain approaches which can handle spatial extent differences and minor illumination changes [7], and autoregressive and other parametric texture modeling techniques [6, 1] which can be computationally efficient. Our approach to texture-based

¹Current Address: Xerox Webster Research Center, 800 Phillips Road, Webster NY 14580.

model-driven selection, as we will see next, combines the advantages of Fourier domain and parametric methods of texture analysis.

2. Representation of Model Texture

The texture or pattern on the model object is assumed to be a planar patch whose projection into the continuous image plane (orthographic projection)² is represented by a continuous signal $s(t_1, t_2)$. The digitized version of this patch can then be represented by the discrete signal $s(m, n)$ where m and n stand for the two coordinate axes of the discrete image plane and s is the intensity signal at those locations. If the texture region $s(m, n)$ is obtained by using a sampling rate much above the Nyquist rate, then it is well-known that its samples are dependent on each other. One way to model this dependency is to say that the signal sample at a location can be predicted from its neighboring samples in a linear predictive fashion. That is, the 2D texture region $s(m, n)$ can be modeled as the output of a 2D linear system with some unknown input $u(m, n)$ so that the texture can be characterized by the impulse response of this linear system as

$$s(m, n) = \sum_{k=-P_{(k,l) \neq (0,0)}}^P \sum_{l=-Q}^Q a_{kl} s(m-k, n-l) + Gu(m, n) \quad (1)$$

where a_{kl} are the coefficients of the impulse response of the linear system and are called the autoregressive (AR) model parameters or coefficients. G is the gain of the system, and $(2P+1) \times (2Q+1) - 1$ represent the order of the AR model, i.e., the number of coefficients of the linear system.

Although the AR model is a compact and computationally simple representation of a texture that is also relatively insensitive to minor textural variations and spatial extent differences, it is not a good domain to examine differences between the model texture and a candidate matching texture region. Earlier methods that used this domain to examine textural differences [1] could mainly detect textural difference in microtextures and further, could not account well for variations in illumination and orientation changes. So, following the advantages of Fourier representations mentioned earlier, we examine a frequency domain interpretation of the AR parameters called the linear prediction spectrum. The linear prediction (LP) spectrum is defined as the power spectrum of the transfer function of the linear system represented in equation 1 and is given by [9]

$$L(\omega_1, \omega_2) = \left| \frac{S(\omega_1, \omega_2)}{U(\omega_1, \omega_2)} \right|^2 = \frac{G^2}{\left| 1 - \sum_k \sum_l a_{kl} e^{-j\omega_1 k} e^{-j\omega_2 l} \right|^2} \quad (2)$$

The linear prediction spectrum is known to be an approximation of the power spectrum that is good at high energy peaks than at low energy peaks [9]. Thus comparing the LP spectra of a model texture and a candidate matching texture region amounts to comparing the respective power spectra with the essential information (high energy peaks) highlighted. Also, the LP spectrum is computationally more efficient compared to Fourier representations. The LP spectrum approximates the power spectrum in a ratio sense (by minimizing the ratio of

²The selection method works even when these assumptions are not strictly observed as we will see in the examples.

power spectrum to the LP spectrum)[9]. This results in a smoothed spectrum that can be coarsely sampled. The lower sampling rate can be derived from the definition of LP spectrum in equation (2) which involves computing the Fourier transform of the AR parameters (and there are only $(2P+1) \times (2Q+1)$ of them as compared to $M \times N$ samples in a texture region of size $M \times N$). Finally, as we shall see next, the LP spectrum tracks orientation changes undergone by a texture in a way that allows the matching of texture under these changes. These properties make the LP spectrum an appropriate representation for texture-based selection.

3. Texture Matching using Isolated Image Texture Regions

We now consider the problem of matching a texture patch obtained from one of the 2D views of a 3D model object³ with a given texture region isolated from the image. Using the LP spectrum representation described above, the 2D texture patch can be described by the tuple $\langle a_{Mpq}, L_M(\omega_1, \omega_2) \rangle$. Any texture region in the image can similarly be characterized by the tuple $\langle a_{Ipq}, L_I(\omega_1, \omega_2) \rangle$. Since the matching of model and image texture LP spectra must account for the fact that the pose of the model object in a given scene may not be as it appeared in its original description, we first examine the effect of pose changes on the LP spectrum representation of the model texture.

Since the surface texture on the model object is assumed to be planar, as the object undergoes a 3D linear transformation in space, the image of the surface texture undergoes an affine transformation [5]. It can be easily shown that the Fourier power spectrum and hence the LP spectrum are invariant to translation so that we need to consider those orientation changes of texture that represent a 2D linear transformation.

We now see the effect of linear transformation of a texture on its LP spectrum. Let the image texture region corresponding to the surface texture on the model object be denoted by a continuous 2d signal $s(t_1, t_2)$. Let $M_t \times N_t$ be its size in continuous coordinates. Let its discretized version be denoted by $s(m, n)$ where $s(m, n) = s(mT_1, nT_2)$ with the sampling rate being (T_1, T_2) . Then its sampled size can be denoted by $M \times N$ where $M^*T_1 \geq M_t$ and $N^*T_2 \geq N_t$. Suppose the surface texture patch undergoes an orientation change. If a new image is created of the changed texture, and assuming the sampling rate remains (T_1, T_2) , (which is true if both textures are imaged using the same camera with lens of the same focal length), then the new image texture representing the changed surface texture is denoted by $s'(m_r, n_r)$. Since the effect of 3D linear transformation of surface texture is a 2d linear transformation (translation ignored) in image plane, the coordinates of the changed signal (image texture) given by $(m_r T_1, n_r T_2)$ correspond to some coordinates $(m_0 T_1, n_0 T_2)$ ⁴ as follows:

$$\begin{pmatrix} m_r T_1 \\ n_r T_2 \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} m_0 T_1 \\ n_0 T_2 \end{pmatrix} \quad (3)$$

Using the above relation, the transformed signal $s'(m_r, n_r)$ can be expressed in

³The 2D views constitute the 3D model description in a recognition system.

⁴the notation (m_0, n_0) is used to denote the fact that the corresponding points can be real numbers unlike (m, n) .

terms of the untransformed signal $s(m, n)$ as

$$s'(m_r, n_r) = s(m_0, n_0) \quad (4)$$

An alternate interpretation of the above equation is that the new signal $s'(m_r, n_r)$ is a periodically resampled version of the continuous signal $s(t_1, t_2)$, with the samples located at $(k.d.T_1 - l.b.T_2, -k.c.T_1 + l.a.T_2)$, where $k = \frac{m_r}{a.d-b.c}$, $l = \frac{n_r}{a.d-b.c}$, and $0 \leq m_r < M'$, $0 \leq n_r < N'$.

If we denote the Fourier transforms of the model and the transformed model textures $s(m, n)$ and $s'(m_r, n_r)$ as $F(\omega_1, \omega_2)$ and $F'(\omega_{1r}, \omega_{2r})$ respectively, then [8]

$$F'(\omega_{1r}, \omega_{2r}) = F(\omega_{10}, \omega_{20}) \quad (5)$$

where

$$\begin{pmatrix} \omega_{1r} \\ \omega_{2r} \end{pmatrix} = \left(\begin{pmatrix} a & b \\ c & d \end{pmatrix}^T \right)^{-1} \begin{pmatrix} \omega_{10} \\ \omega_{20} \end{pmatrix}. \quad (6)$$

Similarly, if we denote the linear prediction spectra of the model and the transformed model textures $s(m, n)$ and $s'(m_r, n_r)$ as $L(\omega_1, \omega_2)$ and $L'(\omega_{1r}, \omega_{2r})$ respectively, then [8]

$$L'(\omega_{1r}, \omega_{2r}) = L(\omega_{10}, \omega_{20}) \quad (7)$$

The proof of the above relationships is available in [8] and is skipped here for brevity. The above equalities, however, hold under the assumption that the doubly periodic sampling rate mentioned before is above the Nyquist rate. A case where this breaks down is when large scale changes occur. A reduction in the size of the signal causes an expansion of the Fourier spectrum so that if the sampling rate is kept fixed (using the same camera with the same focal length, say) then the periodicity of the Fourier spectrum remains the same. This may eventually cause aliasing to occur so that the Fourier spectra can no longer be recognized as transformed versions of each other. For such large scale changes, it may not be possible to spot the resemblance between the two textures in the spatial domain either.

We conclude, therefore, that the Fourier transform and the linear prediction spectrum are not invariant to an affine transformation but instead *as the image texture undergoes a linear transformation, they undergo an inverse linear transformation*. This is a generalization of the previously known results on the effect of scaling and 2D (in-plane) rotational changes on the Fourier transform [2].

3.1 Matching of image and model LP spectra

When the image texture region contains an instance of the model's texture (i.e. assuming perfect isolation), then by the above results, its LP spectrum reflects the orientation change. The parameters of such an orientation change given by a, b, c, d in equation (3) can be recovered from the LP spectrum by finding a pair of corresponding frequencies $\{(\omega_{110}, \omega_{120})(\omega_{11r}, \omega_{12r})\}$ and $\{(\omega_{210}, \omega_{220})(\omega_{21r}, \omega_{22r})\}$ between the model and image LP spectra respectively and using equation 6 to solve for the transformation parameters a, b, c, d . To find these pairs of corresponding frequencies, we observe from equations (5) and (7) that as the texture undergoes a

linear transformation, the location of the spectral peaks (in LP or Fourier spectra) changes but their amplitude remains invariant. Since the identity of peaks is retained (and hence their relative ordering by strengths), the two highest energy peaks of the LP spectra can be paired in the ideal case. In practice, however, since the image texture region may not have been perfectly isolated, or because of the illumination changes from model to image, the amplitude of the peaks in the LP (and also Fourier) spectra will be affected differentially, so that the highest energy peaks in the model's LP spectrum may no longer be of highest energy in the image texture's LP spectrum. Thus search may have to be done among the peaks in the LP spectra to find two corresponding pairs of peaks. This requires $O(K^2L^2)$ pairs to be examined where K = number of peaks in the model's LP spectrum, and L = number of peaks in the image region's LP spectrum. Since the LP spectra have only a few peaks, this is still computationally feasible. Once the transformation parameters are obtained, the model and image region's LP spectra can be aligned using these parameters⁵. The correct set of parameters will ideally align all of the model texture's LP spectrum to the windowed region's LP spectrum indicating a match between the two LP spectra. This, however, ignores the effect of sampling in the LP spectra which can affect the accuracy of the transformation parameters and hence the recovered pose. A more detailed discussion of this problem and an approach to overcome this is available in [8]. Also mentioned there is a way of matching LP spectra when spurious or missing peaks occur due to illumination changes.

4. Extraction of Image Texture Regions

The method of matching LP spectra described above required the isolation of an image texture region that contained only the model texture (even if not all of it is isolated). Such regions can be obtained, under some conditions, without an elaborate texture region segmentation of the image. This can be done by moving a window over the image and maintaining overlap between successive window positions as indicated in Figure 1. We now show that under some conditions, it is possible to find a window that spans only the model texture region regardless of the position and orientation of the model texture in the given image. For this, we first consider the case when the model texture in the image appears as a rectangle of size $N_1 \times N_2$. Suppose we choose to analyze the image of dimensions $I_1 \times I_2$ using rectangular windows of size $M_1 \times M_2$ and an overlap of $L_1 \times L_2$ as shown by the rectangles A and B in Figure 1. Suppose the rectangular patch of texture occurs at a location (x,y) in the image as shown by the rectangle C in that figure. Each analysis window starts at a location $(k.L_1, l.L_2)$ where k and l are integers such that $0 \leq k \leq \lfloor \frac{I_1 - M_1}{L_1} \rfloor$ and $0 \leq l \leq \lfloor \frac{I_2 - M_2}{L_2} \rfloor$. A window that spans only the model texture region must satisfy the constraints for its starting location as $k.L_1 \geq x$ and $k.L_1 \leq x + N_1$ and $k.L_1 + M_1 \leq x + N_1$ for the x-coordinate and similarly $l.L_2 \geq y$ and $l.L_2 \leq y + N_2$ and $l.L_2 + M_2 \leq y + N_2$ for the for the y-coordinate (by the rectangle containment rule). These constraints reduce to finding an integer value of k such that $\frac{x}{L_1} \leq k \leq \frac{N_1 + x - M_1}{L_1}$. Such values of k and l exist provided $\frac{N_1 + x - M_1 - x}{L_1} \geq 1$, and $\frac{N_2 + y - M_2 - y}{L_2} \geq 1$, that is, $M_1 + L_1 \leq N_1$ and

⁵The idea of alignment in frequency domain parallels the alignment in space domain proposed earlier[11]

$$M_2 + L_2 \leq N_2.$$

Thus, as long as we choose a window size smaller than that of the texture and an overlap such that $L_1 \leq N_1 - M_1$ and $L_2 \leq N_2 - M_2$, we can scan the image using overlapping windows and find a window that spans a portion of the model texture region.

Next, consider the case of an actual instance of the model texture appearing in the image. We assume for simplicity that the texture on the model object is square-shaped of dimension $N \times N$ ⁶. Then under linear transformation of the model object, the square texture region undergoes an affine transformation (specified by the parameters a, b, c, d , again ignoring translation) to form a parallelogram as shown by the shape marked E in Figure 1. Within this parallelogram, we can inscribe a rectangle of largest area by considering the closest opposite vertices as shown by the rectangle marked F in Figure 1. The dimensions of this rectangle indicated by $N_3 \times N_4$ are a function of the transformation and are given by $N_3 = (a - b)N$ and $N_4 = (d - c)N$. Then by the previous analysis on rectangular texture regions in an image, if the overlapping windows satisfy the constraints $M_1 + L_1 \leq N_3$ and $M_2 + L_2 \leq N_4$, then a window containing only the model texture region will be found. Although the dimensions N_3 and N_4 are unknown, we can place a bound on these values by restricting the allowable transformations on the model object (and hence the model texture) by requiring $(a - b) \geq s$ and $(d - c) \geq s$, where s is chosen to lie between 0 and 1. This bounds the dimensions N_3 and N_4 to be such that $s.N \leq N_3 \leq N$ and $s.N \leq N_4 \leq N$. With this bound on the dimensions, the windowing constraints can be satisfied by choosing the overlap and window dimensions such that

$$M_1 + L_1 \leq s.N \tag{8}$$

$$M_2 + L_2 \leq s.N \tag{9}$$

5. Texture-based Model-driven Selection

We now combine the concepts of moving window analysis, the LP spectrum representation, and the matching of LP spectra described before, to develop a texture-based model-driven selection mechanism as follows. First, the model texture is described by its LP spectrum using a suitable order AR model. Then the dimension N of the largest square region containing the model texture is noted. Next, a bound s is chosen to limit the allowable transformations. The values of s and N are used to decide the window width (M_1, M_2) and the overlap (L_1, L_2) as specified in equation 8 and 9. The image is analyzed by the overlapping windows, and the image region within each analysis window is described by its LP spectrum using the same order AR model as the model texture⁷. The LP spectrum of each window is then tried as a possible match to the model texture's LP spectrum as described in Section 3.1. The best few matches in the image are taken to indicate the possible places where the model texture (and hence the model object itself) could appear in the image.

⁶Restricting to square shape is not critical since a square can be inscribed in any arbitrary-shaped texture region on the model object.

⁷If the image region contains the model texture then for limited scale changes, the neighborhood of pixel dependence in the texture will remain the same, justifying the choice of the same order AR model for the image region

We now illustrate texture-based model-driven selection by an example. Figure 2a shows a view of the model object (cup) and the texture patch on it that serves as model texture is shown in Figure 2b. The LP spectrum (using a 32x32 pt. FFT) of the model texture using a 2x2 non-causal AR model is shown in Figure 2c. Figure 2d shows a scene in which two instances of the model object appear, one of them reflecting a 3d affine transformation and appears at a different imaging distance (small increase in the apparent size of the object). Since the model texture patch was of size 72 x 52 (was later zero-padded to form a square of size 72 x 72), the image was analyzed using windows of size 48 x 32 with an overlap of 24 x 16 (here s was chosen to be 1.0) to satisfy the constraints of equation 8 and 9. The various windows in which a match for the model texture was found are shown in Figure 2e. The LP spectra for some contiguous overlapping window regions that were found to match the model texture's LP spectra are shown in Figures 2f-i. As can be seen from the figure, although the correct matches are found where the model object exists, there are also a few spurious matches. In general, the match measure is designed to tolerate some missing and spurious peaks [8] so that it can occasionally cause some false positives to occur even after the correct alignment is obtained using the method described in Section 3.1. These false positives can usually be eliminated when texture-based selection is used in combination with other cues such as color as we will see in the next section.

6. Recognition in Combination with Selection

We now examine the use of texture-based selection in reducing the search involved in object recognition. We have developed a recognition system based on alignment by the linear combination of views method [11] and integrated it with texture-based selection. Here, the model object is represented by a set of 2d views with full correspondence between them. Corner features extracted from both the model and image were used to perform the alignment and line segment features were used for doing the verification. The search for corresponding alignment features was done using an interpretation tree type search driven from the image features [3]. Texture-based selection is then performed on the image. However, since the model texture region was assumed to be planar, texture-based selection alone is not sufficient for the recognition of a 3d object (at least one of the features needed for an alignment transform must come from outside a planar region) [11]. For this reason, and to prune the false positive matches that texture-based selection may give, we combined texture-based selection with color-based selection as described in [10] and chose the rest of the corresponding features from color-selected regions. Thus the texture regions that lay within the color selected regions only were retained as matches from texture-based selection. Finally, the regions isolated by texture and color-based selection were used for finding the corresponding features for recognition.

Figure 3 illustrates the combined use of color and texture for selecting regions for recognition. Figure 3a shows a model object possessing both color and texture information. The color description used for color-based selection is shown in Figure 3b while the texture patch used for texture-based selection is shown in Figure 3c. Figure 3d shows a scene in which the model object occurs. The result of color-based selection is shown in Figure 3f while the result of texture-

based selection is shown in Figure 3e. The result of the combined use of color and texture-based selection is shown in Figure 3g. As can be seen from this figure, the false match shown in Figure 3e is removed by the use of color information.

To test the contribution of texture-based selection in reducing the search in recognition, we considered several (around 600) random orderings of the list of corner features in the texture regions in such scenes, and recorded the average number of matches that needed to be tried before finding a tuple of features that is part of the overall winning combination (i.e. is successfully verified by the recognition system). These results are shown in the last column of Table 1. Here the number of matches actually explored by the recognition system for finding seven corresponding corner features are listed. As can be seen from the table, the number of matches that are explored in practice are far fewer compared to the case of recognition without any prior selection (shown in Column 8 of Table 1). In fact, compared to the number of matches explored, detailed verification was done for only a few (about a 1000) of the matches. From this we conclude that the combined use of texture and color-based selection can lead to a vast improvement in the performance of a recognition system.

7. Conclusions

In this paper, we have advocated autoregressive modeling of textures and the resulting linear prediction spectrum as an appropriate representation for addressing the problems in texture-based model-driven selection. We have also demonstrated that this selection mechanism can greatly improve the search performance during recognition and can hence serve as useful front end for a recognition system.

References

- [1] P. de Souza. Texture recognition via autoregression. *Pattern Recognition*, 15(6):471–475, 1982.
- [2] R.C. Gonzalez and P. Wintz. *Digital Image Processing*. Reading: Addison-Wesley, 1987.
- [3] W.E.L. Grimson. *Object Recognition by Computer: The Role of Geometric Constraints*. Cambridge: MIT Press, 1990.
- [4] R.M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man and Cybernetics*, 3(6), 1973.
- [5] D.P. Huttenlocher. *Three-dimensional recognition of solid objects from a two-dimensional image*. PhD thesis, Artificial Intelligence Lab, M.I.T., AI-TR-1045, 1988.
- [6] L.R. Kashyap and A. Khotanzad. A model-based method for rotation invariant texture classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(4):472–481, July 1986.
- [7] S-S. Liu and M.E. Jernighan. Texture analysis and discrimination in additive noise. *Computer Vision, Graphics, and Image Processing*, 49:52–67, 1990.
- [8] Tanveer F. Mahmood. *Attentional selection in object recognition*. PhD thesis, Artificial Intelligence Lab, M.I.T., AI-TR-1420, 1993.
- [9] J. Makhoul. Linear prediction : A tutorial review. *Proceedings of the IEEE*, 63(4):561–580, April 1975.

- [10] T.F. Syeda-Mahmood. Data and model-driven selection using color regions. In *Proceedings of the European Conference on Computer Vision*, pages 321–327, 1992.
- [11] S. Ullman and R. Basri. Recognition by linear combination of models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 992–1006, October 1991.

S.No	M	N	M_t	N_t	Color selected corners in		Estimated search No selection	Actual search
					model	image		
1.	114	484	35	38	69	45	1.55×10^{33}	2.37×10^6
2.	96	580	84	74	12	34	1.66×10^{33}	3.67×10^6
3.	96	1401	84	76	12	25	7.96×10^{35}	7.8×10^6
4.	64	256	52	47	12	31	3.17×10^{29}	1.67×10^5
5.	138	392	58	145	80	67	1.35×10^{33}	2.38×10^7

Table 1: Estimated and actual search reduction during recognition using selection. The estimated search is obtained assuming 7 alignment features so that a search of $O(M^7N^7)$ is required in the worst case. The actual search indicated in the last column is obtained by the combined use of texture and color-based selection as described in text.

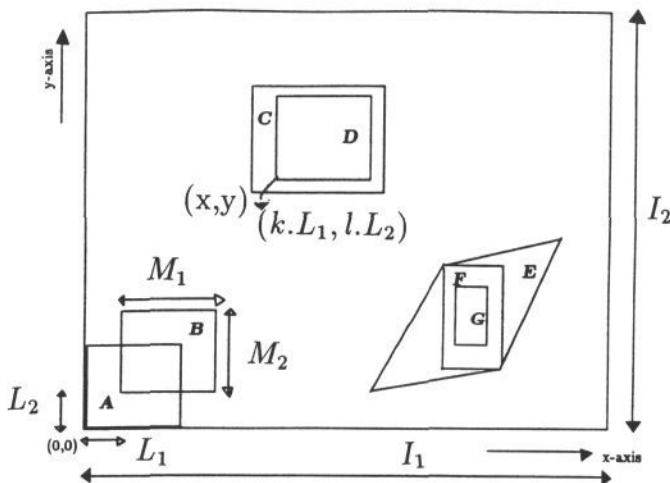


Figure 1: Illustration of overlapping window analysis. The windows labeled A and B are two overlapping windows used to analyze the image. The extent of overlap is indicated by (L_1, L_2) . A rectangle D that spans only the texture region in the rectangle C always exists under the conditions indicated in text. The parallelogram marked E represents the result of a 2D affine transformation on an originally square texture region. The windowed region marked G spanning a portion of E can always be found under the conditions indicated in text.

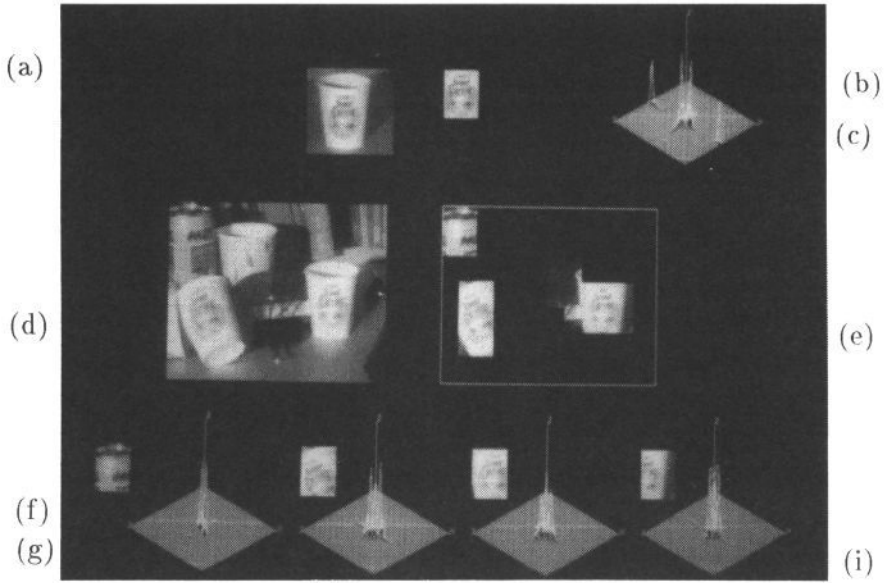


Figure 2: Illustration of model-driven selection using texture. (a) A view of the model object. (b) An extract serving as model texture. (c) The LP spectrum of the model texture using a 2×2 non-causal AR model. (d) A scene in which instances of the model object occur. (e) Result of model texture-based selection. (f) - (i) The LP spectra of some of the windowed regions of (e) that were matched to the model texture by the match measure described in text.



Figure 3: Illustration of the use of color-based selection to reduce false positives in texture-based selection. (a) A view of the model object. (b) Color description of the model. (c) Texture patch on the object used as model texture. (d) A scene in which the object appears. (e) Result of texture-based selection. The false positive match is to be noted here. (f) Result of color-based selection. (g) Result of retaining texture-selected regions that lie within color-selected regions.