Blending Textured Images Using a Non-parametric Multiscale MRF Method

Bernard Tiddeman
School of Computer Science
University of St Andrews
St Andrews, Fife, KY16 9SS
Scotland, UK

bpt@st-and.ac.uk

ABSTRACT

In this paper we describe a new method for improving the representation of textures in blends of multiple images based on a Markov Random Field (MRF) algorithm. We show that direct application of an MRF texture synthesis algorithm across a set of images is unable to capture both the "averageness" of the global image appearance as well as specific textural components. To overcome this problem we vary the width of the Parzen window (used to smooth the conditional probability distribution of the pixel's intensity) as a function of scale, thus making lower pyramid resolutions closer to the Gaussian mean, while maintaining the high resolution textures. We also show that approximating the maxima of the conditional probability distributions with a weighted-average produces very similar results with a significant increase in speed.

Keywords

Facial Prototyping, MRF texture synthesis

1. INTRODUCTION

The ability to construct prototype images (particularly facial images in our own work) has found applications in medicine (e.g. [Tid99]) and psychology (e.g. [Per98]). Prototypes can be used to identify typical characteristics of a given set of images, and can also be used to define transformations on individual images, such as ageing or gender change [Row95]. Prototype images can be created using a combination of warping and blending. Warping is used to align matching features in different images and blending finds the average colour at each spatially aligned pixel. The random nature of surface texture features (e.g. wrinkles, pores or hair in facial images) means that it is impossible to align surface texture components and so information is lost in the blending process.

The analysis and synthesis of textures in images has attracted a large amount of research from a wide range of disciplines. Computer vision researchers are interested in image segmentation and object localisation from textures, psychologists have developed texture analysis and synthesis algorithms to investigate the neurological basis of texture recognition and computer graphics experts are interested in replicating textures for wrapping objects. In our research we are interested in studying facial perception, often using prototypical or "average" face images. Prototypes have previously been created by blending faces together, after normalising the shape to the average using image warping. These "shape and colour" prototypes do not have realistic textural detail i.e. the hair on the top of the head, eyebrows, beards / stubble, wrinkles, pores, spots, moles and liver spots do not have a realistic appearance.

In our previous work aimed at improving the representation of textures in prototype facial images [Tid01] we based the work on the methods of Heger and Bergen [Hee95] and Simomcelli and Portilla [Sim98] who have used properties of the histograms of wavelet subbands to synthesise textures from examples. Because the textures vary across the image we used a local approximation to the shape of the histogram by recording the wavelet magnitude in a small region about each point in each wavelet subband. This method produced prototypes that were perceived as having the correct age - the same as the perceived age of the sample from which it was constructed. Even so, the prototypes still lacked realism, particularly in the more variable textures such as the hair.

Journal of WSCG, Vol.12, No.1-3, ISSN 1213-6972
WSCG'2004, February 2-6, 2003, Plzen, Czech Republic.

In this work we take a completely different approach, basing it instead on Markov Random Field (MRF) methods. Many of the proposed methods for texture synthesis-by-analysis (e.g. Efros and Leung's method [Efr99]) can be seen as approximations to sampling from a probability distribution for the pixel intensity. The probability distribution for each pixel is conditional on the values in some neighbourhood (a region of the pixels around the point) of the pixel. By sampling from this distribution at each pixel a copy of the original texture can be created.

Multiscale MRF based algorithms and approximations to them have been shown to improve both the quality and the speed of texture synthesis. The quality is improved because the characteristic scale(s) of the texture are unknown, and the multiscale approach covers all scales from low to high resolution. The speed is improved because a large high-resolution neighbourhood is built up from many smaller neighbourhoods at different resolutions.

The shape of the neighbourhood effects the speed and quality of reconstruction. The most realistic assumption is that a pixel's neighbourhood is symmetrical, but this acausal assumption requires the use of slow iterative optimisation. In this work we use a *causal* neighbourhood consisting of a non-symmetrical half-plane (NSHP) neighbourhood at the current resolutions and a square neighbourhood at the previous (coarser) resolution. The use of a symmetrical low-resolution neighbourhood in addition to the NSHP high-resolution neighbourhood helps to stabilise the reconstruction, without the need for optimising the probabilities of the all the pixels simultaneously.

Because the textures vary across the facial images, prototyping multiple face images requires a different approach than synthesising a new example of a single texture. In this work we build the conditional probability distribution by sampling from the same spatial location across multiple images, rather than sampling from multiple locations in a single texture image. In addition, rather than randomly sampling from this distribution, we select the most probable greyscale value.

As we will see in the following sections, simply picking the most likely pixel from the probability distribution is not sufficient to synthesise prototypical images, because the texture "locks" on to a single image at a low resolution. Instead we use a method for maintaining "averageness" at low resolutions, but selecting increasingly specific texture components as the resolution increases.

In the remainder of this paper we first describe our new technique for synthesising images of a single example texture and demonstrate its effectiveness on a number of real and synthetic textures. We then present the method modified to blend images with spatially varying textures. Finally we present the results of some experiments to demonstrate the effectiveness of the new algorithms.

2. LITERATURE REVIEW

Image Prototyping Literature

Several methods have been proposed for the construction of prototype images. The earliest simply cross-dissolved a sample of facial images after registering the eye centres and mouth [Gal78][Lan90]. This technique produces facial prototypes that become blurred with increasing distance from the eyes and mouth because facial features are not properly aligned before their combination. Distorting the images in the sample using image warping [Wol90][Rup95] can align corresponding features leading to sharper prototypes [Ben93]. Corresponding features can be labelled manually [Ben93][Bre85] or recovered automatically [Yui92][Coo95][Bey96][Vet97].

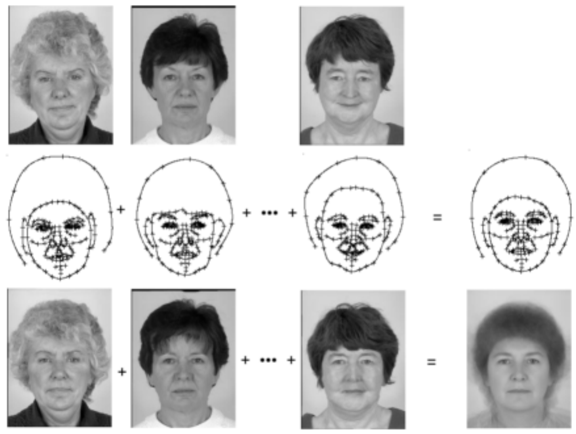

Figure 1. Basic image averaging. The input images (top row) are delineated (middle row) and the average shape is calculated from the delineated feature points (middle-right). The original images are then warped into the average shape (bottom row) and the average colour is found at each pixel (bottom-right).

In this paper we extend the prototyping method of Benson and Perrett [Ben93] (Figure 1). First, delineation of salient features with points and lines is performed by hand or automatically using active shape models [Coo95]. The average shape is defined as the average positions of each of these points. Each image is then warped into the average shape using one of the many available methods. Averaging the colour at each pixel across the shape normalised component images produces the prototypes that will be referred to here as ‘untextured prototypes’.

Texture synthesis-by-analysis literature

The problem tackled by most texture synthesis-by-analysis methods can be stated as:

Given an example image that is perceived as a single texture (e.g. grass, pebbles or wood), produce a non-identical image that is perceptually of the same texture.

This is useful in areas such as 3D modelling to create large texture patches for wrapping around complex 3D objects. The problem tackled in this paper is somewhat different, i.e.

Given a set of spatially normalised images containing a number of different textures (e.g. hair, wrinkles and stubble) construct a prototype with perceptually representative textures.

Nonetheless, the large body of literature on texture synthesis-by-analysis provides many possible methods that could be adapted for the problem of creating appropriate textures in prototype images. The most popular methods for modelling natural textures are MRF-based methods [Efr99] [Che85a] [Cro83] [Che85b] [Pag98] [Has81], wavelet-based methods [Hee95] [Sim98] or a combination of the two [DeB97][Zha98].

Julesz was among the first to suggest a statistical model of texture based on N^{th} order pixel statistics [Jul62]. MRF texture models are statistical models that assume that the probability of a pixel having a particular intensity is dependent only on the intensities of the neighbouring pixels. The probability distribution for a pixel, given the intensities of its neighbours, can be calculated from the sample texture. For example, a parametric Gaussian mixture model or a non-parametric model such as the Parzen window method can be used to estimate the probability density function. The task of image synthesis is then to optimise the probabilities across the synthetic image to match the probability distributions of the sample or training image. This is often achieved by using an iterative scheme, for example by simulated annealing.

The computational complexity of the MRF methods increases with the size of the neighbourhood considered. Therefore, in order to represent large as well as small-scale features accurately and efficiently, multiscale MRF methods have been proposed. Even so, the iterative nature of the image synthesis makes MRF methods prohibitively slow given current computing. An alternative speed enhancement adopted by several algorithms approximate the local conditional probability distribution function (LCPDF) using stochastic sampling, selecting the k nearest matches and selecting a texture element (i.e. a pixel) from among them. These methods are usually implemented using a causal neighbourhood, thus eliminating the need to use slow iterative methods, at the possible expense of accuracy and stability in the reconstruction.

An alternative to iterative MRF optimisation is to decompose the image using basis functions that give it a simpler statistical description. In wavelet analysis an image is decomposed using basis functions that are well localised in both the spatial and frequency domains. This means that the wavelet analysis can represent efficiently both high and low frequency texture components without losing spatial localisation. In addition, wavelet style decomposition is thought to play an important part in the early stages of visual processing in mammals [Dau80] [Mar82].

Wavelet-based methods have been used to synthesise textured images from a relatively small number of parameters from the source texture, such as correlations within and between different subbands. In this work we are not trying to synthesise a single texture image so there is no advantage to having a concise parametric texture model i.e. we would require a different set of texture parameters at each pixel, rather than a single set for the whole image. Therefore wavelet-based methods are unlikely to offer a significant time or space complexity improvement for blending textured images over direct MRF synthesis. It is possible that wavelet-based methods might give a quality improvement, but we leave this as a topic for future research.

3. METHOD

Single Texture Synthesis

Our method for synthesising textures is based on the non-causal non-parametric multiscale method of [Pag98] with some modifications, most notably that it is a causal variation of the original algorithm. The algorithm starts by making a low-resolution approximation to the texture by randomly sampling from the low-resolution version of the example texture. This could be as small as a single pixel. Each successively finer resolution version of the texture is then built up by iteratively optimising each pixel's probability by sampling from the local conditional probability distribution for the pixel given the greyscale values in a local neighbourhood of the pixel.

One possible drawback to the original algorithm is that the multiresolution pyramid was created by simply subsampling the example texture at each scale. The next finer resolution image was then initialised by expanding the coarser scale image by pixel doubling. During the MRF optimisation the greyscales of the even indexed pixels were fixed to the values calculated at the coarser resolution. This use of subsampling could lead to aliasing of the textures at coarser resolutions. Instead we low-pass filter the image before subsampling. When the multiresolution pyramid is constructed with low-pass filtering the even pixel greyscales are no longer valid estimates of the finer resolution pixel values.

Instead of using a non-causal neighbourhood we use a causal neighbourhood that includes information from both the current scale and the previous resolution scale. A square neighbourhood surrounding the pixel at the coarser resolution is combined with a NSHP neighbourhood at the current scale. Hence our algorithm is *causal* i.e. the next pixel's value is determined by its neighbourhood, but the pixels in the neighbourhood are not dependent on the current pixel. The use of an accompanying low-resolution neighbourhood helps to stabilise the reconstruction and eliminates the need for very slow iterative methods such as simulated annealing.

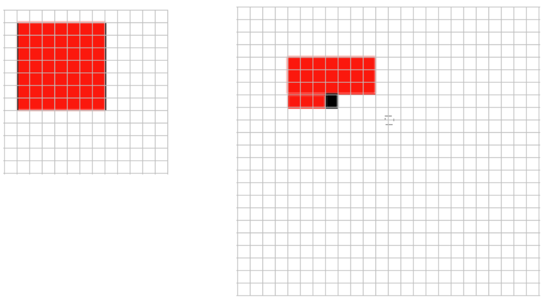

Figure 2. The pixel neighbourhoods used in this paper span two levels in the multiresolution pyramid. On the left is the square neighbourhood in the coarse resolution approximation to the solution that has already been calculated. On the right is the NSHP neighbourhood at the current resolution. This image has been calculated in scanline order up to the pixel we are trying to estimate (shown in black).

At each pixel the conditional probability distribution is estimated from the example image using a Parzen window method (Algorithm 1). The probability distribution can be thought of as a 1D "slice" through an N-dimensional distribution, where there is one dimension for each pixel in the neighbourhood. The non-parametric distribution is estimated by smoothing the N-dimensional histogram using a kernel function. We use the multidimensional Gaussian function as the kernel. The shape of the distribution is also critically dependent on the kernel's smoothing parameter, h, in this case the width of the Gaussian. The "optimal" value, h_0, given in [Pag98] for the smoothing parameter is only optimal if the true underlying distribution is Gaussian [Sil86]. Experiments using the Gaussian assumption for single texture synthesis proved unsuccessful in many cases because the distribution is smoothed too much. (Figure 3). This may be because of the high degree of correlation between the same pixels at neighbouring resolutions.

Approximations to the MRF method such as [Efr99] select randomly from the samples having the closest match to the neighbourhood i.e. they will never choose an intensity or wavelet coefficient that is not present in the original image. This behaviour is approximately equivalent to choosing a relatively small value for the smoothing parameter. We have experimented with various multiples of h_0 and have found that h =0.25 h_0 produces good results with a range of textures. We use a 5 by 5 pixel neighbourhood at the coarser resolution combined with a 7 by 4 NSHP neighbourhood at the current resolution about the position of the target pixel.

Algorithm 1: Calculate LCPDF at output pixel (x,y)
inputs:
Array of source images S, destination image D, smoothing parameter h, sample size M
begin:
1. Create array p of length L (the number of greyscales) and initialise to 0
2. Array u = values of pixels in neighbourhood N of (x,y) in D
3. For each example image k = 0 to M
    3.1 Array v = values of pixels in neighbourhood N of (x,y) in image S[k]
    3.2 p[S[k](X,Y)] += Gaussian(v, w, h)
4. Smooth p with 1D Gaussian of width h and re-normalise
5. Return probability distribution p
end

It should be noted that the use of this algorithm in the form presented here is extremely slow because of the need to sample the entire image to build up an accurate LCPDF for each pixel at each scale. Optimisations are possible e.g. by using a k-nearest neighbours approach, but these are not applicable to the case of blending multiple textured images. This is because the sample size is typically much smaller i.e. the number of images passed to the blending algorithm is far less than the number of pixels in the single texture synthesis algorithm.

Blending Textured Images

We can extend the MRF single texture synthesis method described above to blending textured images by estimating the conditional probability density at a point by sampling from a fixed location across multiple images rather than from different locations in a single image. We then have the option of sampling randomly from this distribution, which would result in a typical, if not necessarily prototypical, facial image from the set. An alternative is to pick the highest probability greyscale at each pixel, which should lead to a prototypical image. We have tried this algorithm with both real and synthetic images containing multiple textures, in which each texture to be blended is aligned with a different example of the same texture. This method appears to work well, but there is a problem in that it often "locks" on to a single image at a low resolution resulting in a final image that is a patchwork of areas copied from a small number of different images in the sample.

In order to improve on this method we have experimented with varying the Parzen window width as a function of scale. We choose a function that switches smoothly from a wide to a narrow smoothing window as the level decreases. By smoothing the distribution with a wider Gaussian at low-resolutions we force the overall appearance of the prototype to be closer to the original average. As the scale decreases we use a narrower Gaussian to allow the selection of more specific texture components at the finer scales.

In this paper we use a linear function of the pyramid level to scale the width of the smoothing parameter. We choose h to be the same as in the single texture synthesis examples (i.e. 0.25 h_0) at the finest scale to ensure that the textures are accurately reproduced in the final image. We then use a linearly increasing function of scale with level i.e. $h_l = (0.25+\alpha\ l)\ h_0$, where l is the current pyramid level (starting at 0) and α is the rate of increase of the smoothing parameter. We have found that large values of α can destroy the continuity of the more structured textures, such as the hair. In this paper we use $\alpha = 0.5$ as this appears to give a good trade off between average appearance (i.e. similarity to the untextured averages) and texture quality. More systematic selection of α will require a range of perceptual experiments that we leave as a topic for future experimentation.

The above method is also rather slow because of the need to smooth the 1-D distribution at each pixel, in order to build up the non-parametric probability distribution. As an approximation to this method we have experimented with using a weighted-average approach, where the weightings are chosen so that images with a more similar neighbourhood are given a higher weight (Algorithm 2). The weightings we use are equivalent to assuming that the peak of the non-parametric LCPDF is the same as its Gaussian mean.

Algorithm 2: Estimate greyscale mean of LCPDF at output pixel (x,y)
inputs:
Array of source images S, destination image D, smoothing parameter h, sample size M
begin:
1. Initialise variables sum=0 and weight=0
2. Array u = values of pixels in neighbourhood N of (x,y) in D
3. For each example image k = 0 to M
  3.1 Array v = values of pixels in neighbourhood N of (x,y) in image S[k]
  3.2 P = Gaussian(v, w, h)
  *3.3 sum += P*S[k](x,y)*
  3.4 weight += P
4. sum = sum/weight
5. Return sum
end

4. RESULTS

Figure 3 shows the results of synthesising patches of singles textures from a single example texture. The choice of a narrow Parzen smoothing window clearly gives superior results in these examples. Figure 4 shows the results of blending an image consisting of 4 different textures using no texture processing, wavelet-based texture processing, full MRF-based texture processing and the approximate MRF method based on weighted-averaging. The two MRF based methods appear to reproduce the textures more faithfully than either no texture processing or the wavelet magnitude method.

The problem of texture "locking" is demonstrated in figure 5. The MRF blend with $h = 0.25\ h_0$ is clearly just a patchwork of examples from the set. Increasing the smoothing parameter (to $h = 2.5\ h_0$) at all scales does not help as the textures become blurred, even when the global appearance becomes more "average". Varying the Parzen window width as described in the text can retain the global averageness as well as realistic textures.

Figure 6 shows some further examples of male and female facial prototypes of different ages together with the untextured prototypes and the wavelet-based textured prototypes. Both the full MRF and the Gaussian approximation produce more realistic results than the untextured or wavelet processed versions.

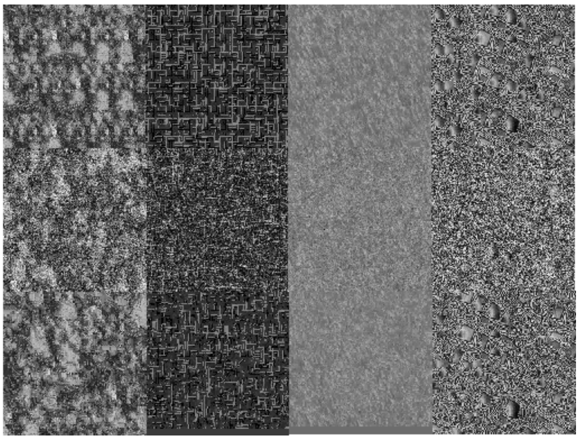

Figure 3. Single texture synthesis. Top row: Original textures. Second row: MRF synthesised using the Gaussian "optimal" smoothing width. Bottom row: MRF synthesised using one quarter the Gaussian "optimal" smoothing width.

5. CONCLUSIONS

We have demonstrated the effectiveness of applying MRF texture algorithms to the problem of creating prototypes from collections of images. We have discovered the problem of textures "locking" when using a straightforward application of the original algorithm to multiple images, and have solved it by increasing the Parzen window width as a function of spatial scale. In addition we have shown that the Gaussian approximation to the LCPDF (i.e. the weighted average approach) achieves results comparable to the full non-parametric MRF method. The results show that the adapted MRF texture method works for blending multiple images and has produced highly realistic and plausibly "average" images. Our next task is to validate both the realism and the prototypicality of the new prototypes using perceptual experiments. We will also attempt to adapt our existing transformation methods, for example prototype-based facial ageing [Row98][Tid01], to use the new texture model.

6. ACKNOWLEDGMENTS

I would like to thank Unilever US for funding this research and members of the Perception Laboratory, Psychology, University of St Andrews for useful comments on this work.

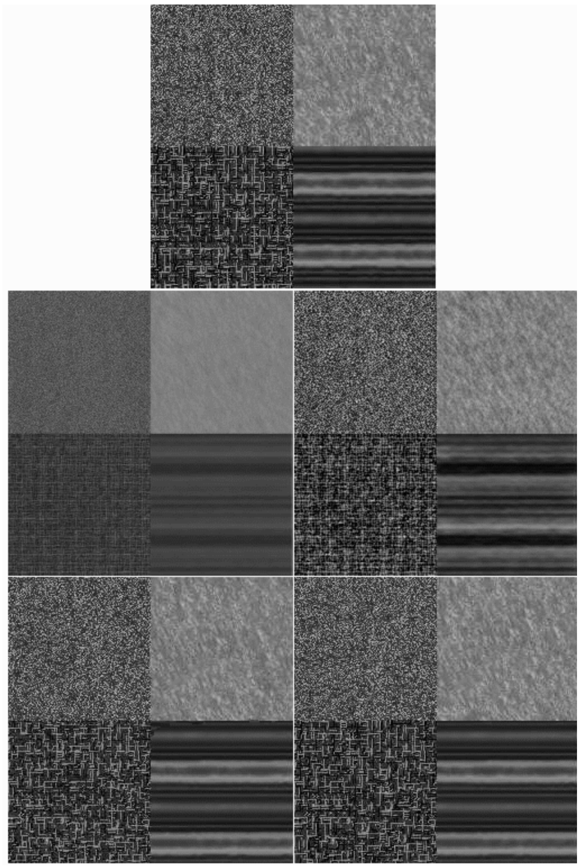

Figure 4. Multi texture-blending examples. Top: an example of the one of the input images from the set, all of the images contained the same texure in each corner shifted by a random amount. Centre left: Blending of the 17 images without texture processing. Centre right: Blending of the images using the wavelet-based method of [1]. Bottom left: Blending using the full MRF method. Bottom right: Blending using the weighted average approach.

Figure 5. Effect of varying the width, h, of the Parzen window. Left: h= 0.25 h_0, the image has large areas that are just direct copies of individual's in the sample. Centre: h= 2.5 h_0, the image has burred textures, but an overall appearance closer to the expected mean. Right: h= $(0.25 + 0.5\ l)h_0$ varying h with image pyramid level, l, produces a good mixture of global appearance and appropriate textures.

Figure 6. Prototype male and female face examples of two age groups. First column - untextured blends. Second column - wavelet processed blends. Third column - full MRF textured blends. Fourth column - weighted average approximate MRF method.

7. REFERENCES

[Tid01] B. Tiddeman, G. Rabey and N. Duffy, "Synthesis and Transformation of Three-dimensional Facial Images", *IEEE Engineering in Medicine and Biology*, Vol. 18, No. 6, pp 64-69, November/December 1999.

[Per98] D.I. Perrett, K.J. Less, I. Penton Voak, D. Rowland, S. Yoshikawa, D.M. Burt, S.P. Henzil, D.L. Castles, and S. Akamatsu "Effects of sexual dimorphism on facial attractiveness." *Nature*, 394, 884-887, 1998.

[Row95] D. A. Rowland and D. I. Perrett, "Manipulating Facial Appearance Through Shape and Color", *IEEE Computer Graphics and Applications*, Vol. 15, No. 5, pp70-76, 1995.

[Tid01] B. Tiddeman, D.M. Burt and D. Perrett, "Prototyping and transforming facial textures for perception research", *IEEE Computer Graphics and Applications*, pp 42-50, Sept/Oct 2001.

[Hee95] D.J. Heeger and J.R. Bergen, "Pyramid-based texture analysis/synthesis," *Proceedings SIGGRAPH '95*, Los Angeles CA, pp 229-238, 1995.

[Sim98] E.P. Simoncelli and J. Portilla, "Texture characterization via joint statistics of wavelet coefficient magnitudes," *Proceedings of Fifth International Conference on Image Processing*, Vol. 1, Chicago, IL, October 1998.

[Efr99] A. Efros and T. Leung, "Texture synthesis by non-parametric sampling", in *Proceedings 7th International Conference on Computer Vision*, pp 1033-1038, 1999.

[Gal78] F.J. Galton, "Composite Portraits," *Nature*, Vol. 18, pp97-100, 1878.

[Lan90] J.H. Langlois and L.A. Roggman, "Attractive faces are only average," *Psychological Science*, Vol. 1, pp. 115-121, 1990.

[Wol90] G. Wolberg, *Digital image warping*, IEEE Computer Society Press, Los Alamitos (CA), 1990.

[Rup95] D. Ruprecht and H. Muller, "Image warping with scattered data interpolation," *IEEE Computer Graphics and Applications*, Vol. 15, No. 2, pp. 37-43, 1995.

[Ben93] P.J. Benson and D.I. Perrett "Extracting prototypical facial images from exemplars," *Perception*, Vol. 22, pp 257-262, 1993.

[Bre95] S.E.Brennan, "The character generator," *Leonardo*, Vol. 18, pp. 170-178, 1985.

[Yui92] A.L. Yuille, P.W. Hallinan and D.S. Cohen, "Feature extraction from faces using deformable models," *Int. Journal of Computer Vision*, Vol. 8, pp. 99-111, 1992.

[Coo95] T.F.Cootes, C.J. Taylor and A. Lanatis, "Active shape models – their training and application," *Computer Vision Graphics and Image Understanding*, Vol. 61, pp. 38-59, 1995.

[Bey96] D. Beymer and T. Poggio, "Image reresentation for visual learning," *Science*, Vol. 272, pp. 1905-1909, 1996.

[Vet97] T. Vetter and T. Poggio, ""Linear object classes and image synthesis from a single example image," IEEE Trans. on PAMI, Vol. 19, pp 733-742, 1997.

[Che85a] R. Chellappa and S. Chatterjee, "Texture classification using GMRF models," *IEEE Trans. Acoustics, Speech and Signal Processing*, Vol. 33, pp 959-963, 1985.

[Cro83] G.R. Cross and A.K. Jain, "Markov random field texture models*," IEEE Trans. on PAMI*, Vol. 5, No. 1, pp 25-39, 1983.

[Che85b] R. Chellappa and S. Chatterjee, "Texture synthesis using 2-D noncausal autoregressive models," *IEEE Trans. on Acoustics, Speech and Signal Processing*, Vol. 33, pp 194-203, 1985.

[Pag98] R. Paget and I.D. Longstaff, "Texture synthesis via a noncausal nonparametric multiscale Markov random field," *IEEE Transactions on Image Processing*, Vol. 7, pp 925-931, 1998.

[Has81] M. Hassner and J. Sklansky, "The use of Markov random fields as models of texture," in *Image Modelling*, A. Rosenfeld ed., Academic Press (New York), 1981.

[DeB97] J.S. De Bonet, "Multiresolution sampling procedure for analysis and synthesis of texture images," *Proceedings SIGGRAPH '97*, Los Angeles CA, pp 361-368, 1997.

[Zha98] J. Zhang, D. Wang and Q.N. Tran, "A wavelet-based multiresolution statistical model for texture*," IEEE Trans. on Image Processing*, Vol. 7, pp 1621-1627, 1998.

[Jul62] B. Julesz, "Visual pattern discrimination," *IRE Transactions on Information Theory*, IT-8, 1962.

[Dau80] J.G. Daugman, Vision Research, Vol. 20, pp 847-856, 1980.

[Mar82] D. Marr, Vision, W. H. Freeman (San Fransisco), 1982.

[Sil86] B.W.Silverman, *Density Estimation for Statistics and Data Analysis*, Chapman and Hall Ltd, 1986.