

Enhancing Videoconferencing Using Spatially Varying Sensing

Anup Basu, *Member, IEEE*, and Kevin James Wiebe

Abstract—Human vision can be characterized as a variable resolution system—the region around the fovea (point of attention) is observed with great detail, whereas the periphery is viewed in lesser detail. In this work, we show that spatially varying sensing (resembling the human eye) can indeed be useful in videoconferencing. A system that can incorporate multiple and moving foveae is described. There are various possible ways of implementing multiple foveae and combining information from them. Some of these alternative strategies are discussed and results are compared. We also show the advantage of using spatially varying sensing as a preprocessor to JPEG. The methods described here can be useful in designing teleconferencing systems and image databases.

I. INTRODUCTION

THE ability of computers to display and manipulate images has greatly increased since the introduction of the desktop computer. Computers are faster, screens have higher resolution, and costs have been reduced. The advent of multimedia has expanded the use of both still and moving images. These advances, however, have been accompanied by certain problems, not the least of which is the size of the typical image. Simply put, images contain very large amounts of information. Some form of compression is necessary in order to reduce the bandwidth needed for transmission, the storage memory, and the processing required for manipulation.

Data compression techniques have existed since the early days of computing science. However, initial schemes were aimed mainly at the compression of text and/or programs [22], not at images. Image compression requires new methods and ideas which may not apply to other data types, and the range of uses for image compression within multimedia systems is vast. Methods used to compress a single, still image can be used in the publishing industry or in an image archival system. The compression of moving pictures can be used in teleconferencing (videophones), electronic storage of movies, or even high-definition television (HDTV) transmission.

Compression methods can be broadly classified as either lossless or lossy, depending on the method's effect on the data being compressed. In lossless methods, data is unaffected by compression. That is, the data after compression is identical to the data before compression. (The data may be stored in

a different form, but exact pixel values remain unchanged.) In contrast, a lossy method alters the data. Generally, lossy schemes produce better *compression ratios* (reduction in the number of bytes required to represent data) than lossless methods.

In many images, there exists one or more areas which are of greater interest than the remainder of the picture. In such an area (fovea) more detail is required. The outer regions (periphery) are often of secondary importance, and thus less detail is needed. This decrease in detail within the periphery may be achieved by spatial subsampling. The concept of variable resolution (VR) has been applied to many different areas [27], [34], [35], but its application to image compression is relatively new. There are a number of distinct advantages of variable resolution compression:

- guaranteed compression ratios can be obtained by controlling the sampling;
- the VR transform can be achieved using a look-up table, thus VR compression can be performed quickly;
- images compressed using the VR transform can be further compressed using other methods [3], [15], [36], [38].

Clearly, VR cannot be used to compress all data, since there are many cases where the areas of primary interest cannot be determined in advance (such as medical images); and where the distortions caused may be unacceptable. However, there are some applications (such as videoconferencing) where this technique gives high compression ratios for relatively low computational cost.

II. BACKGROUND

The roots of data compression lie in entropy encoding techniques. The purpose of these techniques is to alter the bit stream so that the same information is represented using fewer bytes. These schemes range from minimum redundancy codes (such as Huffman encoding) to dictionary coding methods (LZ77, LZ78, and LZW) [39].

The ability of any entropy encoding technique to compress data depends on the presence of patterns (repeating bytes or skewed frequency distributions) within the input data stream.

A number of different methods have been applied to the special problems encountered in image compression. Vector quantizing, wavelets and fractal-based compression have all been used, with varying degrees of success [15].

Much attention has been given to compression schemes based on the discrete cosine transform (DCT) [38]. The DCT is able to convert images from the spatial to the frequency

Manuscript received April 24, 1994; revised November 9, 1995 and May 25, 1997. This work was supported in part by the Canadian Natural Sciences and Engineering Research Council. Parts of this work were presented in the IAPR/IEEE International Conference on Pattern Recognition, Jerusalem, Israel, Oct. 1994.

The authors are with the Department of Computing Science, University of Alberta, Edmonton, Alta., Canada T6G 2H1 (e-mail: anup@cs.ualberta.ca).

Publisher Item Identifier S 1083-4427(98)01578-1.

domain. Once converted, high frequencies (ones which the human eye cannot detect) can be eliminated. One advantage is that compression and quality can be traded against each other. If higher compression is required, additional high frequencies can be eliminated at a corresponding decrease in quality. The DCT is the basis of a lossy still image compression scheme developed by the Joint Photographic Expert Group (JPEG). JPEG allows quality to be traded-off for compression, usually by specifying a quality value.

Several standards have been developed for motion pictures. The CCITT standard H.261 (also known as p*64) was developed for use in visual telephones transmitting over ISDN networks [19]. More general in nature is a standard developed by the Motion Picture Expert Group (MPEG) [12]. Both the p*64 and MPEG standards use the DCT, but both also use other methods which help remove temporal redundancy. In addition to the above methods, variable bit rate coding techniques have been used to take advantage of the burstiness of video sources and statistical multiplexing capability of asynchronous transfer mode (ATM) networks [11], [29], [37]. Other researchers have looked into the use of subband coding [10], [18], [25], [30] and wavelets for image compression [2], [3], [8], [40].

One of the prime applications for motion pictures has been videoconferencing, where individuals can communicate with each other visually over networks as easily as using a telephone [5], [7], [17]. Early attempts to create digital videoconferencing systems were often limited in their capabilities [1]. However, videoconferencing systems, often allowing the integration of voice, video, and text, have recently begun to make their appearance on desktop workstations and PC's within "multimedia" systems. These systems may offer support for real-time remote collaborative work, and make good use of the bandwidth available with new networking technology. However, compression is required for video transmission across limited bandwidth channels.

The next major step in digital compression will involve the replacement of current analog television signals with higher resolution digital signals. This may be combined with upgrades of cable television networks. Decisions on standards for high-definition television (HDTV) are currently being made.

III. THE VR TRANSFORM

Studies have shown that animate vision has much higher resolution in the center of the visual field than in the periphery [9], [13], [14], [16], [20], [21], [23], [31], [32]. Several mathematical models and vision systems have been developed based on this variable resolution concept [4], [6], [24], [26], [28], [33]. More recently, cameras have been developed which are capable of capturing images using log-polar coordinates directly, thereby bypassing the need for mathematical transformations altogether in some cases.

Although computers may not approach the ability of animate vision, such biological systems can serve as a model for transforming images when a scene contains regions of varying importance. These transformations can be performed using special camera lenses (e.g., "fisheye" lenses) or by the use of

mathematical models, such as those developed by Swartz [28]. Results presented here use a simplified mathematical model to implement the VR transform.

A. The Basic Variable Resolution Model

The VR transform used here [4] has two parameters which affect the resultant image: the expected savings (compression), and alpha (α), which controls the distortion at the edges of the image with respect to the fovea. A high α value gives a sharply defined fovea with a poorly defined periphery; a small α value makes the fovea and periphery closer in resolution. Under the VR transform, a pixel with polar coordinates (r, θ) is mapped to (v, θ) where

$$v = \ln(r * \alpha + 1) * s. \quad (1)$$

In other words, the pixel is moved from r to v units away from the fovea. This transformation is easily reversed, allowing r to be defined in terms of v

$$r = \frac{\exp(v/s) - 1}{\alpha}. \quad (2)$$

The value s is a scaling factor used to control the overall compression ratio. It is calculated so that points at the maximum distance from the fovea in the original image are at the maximum possible distance in the VR image

$$s = \frac{v_{\max}}{\ln(r_{\max} * \alpha + 1)}. \quad (3)$$

It must be noted that pixels are discrete elements. Thus, when reducing the size of the image via the VR transform, one VR pixel may represent several pixels in the original image.

B. Extending the Model

There is one difficulty with the basic variable resolution model and its application to image compression. Under the VR transform, images are not rectangular. If storage in a rectangular field is necessary, areas of the image must be clipped off, or the image will have unused pixels in the corners. The problem is magnified with high α values and when the fovea is not located in the center of an image.

One approach to solving this problem uses multiple scaling factors, each scaling factor dependent on the angle θ in polar coordinates ("modified variable resolution"). Each angle θ has its own maximum distances to the image edge, and thus its own scaling factor. The transformed image can now be mapped to a rectangle with full space utilization, regardless of the location of the fovea. This method does a reasonable job of maintaining the isotropic properties of the original formulae, but is relatively complex.

Another approach to dealing with nonrectangular compressed images is to greatly simplify the formulae by isolating the vertical and horizontal components ["Cartesian variable resolution" (CVR)]. Figs. 1–3 demonstrate the alternative approaches. Computational complexity is significantly decreased for the CVR method—however, isotropic accuracy is reduced as well. For a given image with the fovea located at (x_0, y_0) , for every pixel (x, y) in the original image, we define the



Fig. 1. Original image.

distance from (x, y) in x and y directions as d_x and d_y , respectively, from the following equations:

$$d_x = x - x_0 \quad (4)$$

$$d_y = y - y_0. \quad (5)$$

So, (x, y) is mapped to point (x_1, y_1) where

$$x_1 = x_0 + \ln(dx * \alpha + 1) * s_x \quad (6)$$

$$y_1 = y_0 + \ln(dy * \alpha + 1) * s_y. \quad (7)$$

In other words, here a pixel is moved from d_x and d_y to d_{vx} and d_{vy} units away from the fovea in x and y directions, where

$$d_{vx} = \ln(dx * \alpha + 1) * s_x \quad (8)$$

$$d_{vy} = \ln(dy * \alpha + 1) * s_y. \quad (9)$$

This transformation can be easily reversed, allowing d_x and d_y to be defined in terms of d_{vx} and d_{vy}

$$d_x = \frac{\exp(d_{vx}/s_x) - 1}{\alpha} \quad (10)$$

$$d_y = \frac{\exp(d_{vy}/s_y) - 1}{\alpha}. \quad (11)$$

As in the basic VR model, the values s_x and s_y are scaling factors used to control the overall compression ratio. The scaling factor is calculated so that points at the maximum x or y distance from the fovea in the original image are at the given maximum possible x and y distance in the compressed image

$$s_x = \frac{d_{vx_{max}}}{\ln(d_{x_{max}} * \alpha + 1)} \quad (12)$$

$$s_y = \frac{d_{vy_{max}}}{\ln(d_{y_{max}} * \alpha + 1)}. \quad (13)$$

As with the modified VR method, different scaling factors can be used. The CVR method can vary the scaling factors, both horizontally and vertically, depending on the position of the

fovea. The lookup tables constructed using this method will thus be dependent on the position of fovea, but the resulting compressed images will be of constant size. If, however, the scaling factors used vertically and horizontally are computed independently from the position of the fovea the compressed images will vary in size depending on the fovea location. The advantage to this method is that the tables need not be recomputed each time the fovea changes location. This is important when implementing a *moving fovea*.

C. Varying Distortion and Scaling Factors for Optimal Effect

Cartesian VR transform has another advantage. Separation of s_x and s_y allows us to achieve optimal effects in maintaining the area of maximum resolution, independent of the absolute or relative locations of multiple foveae. Having separate s_x and s_y for all angles (θ) suggests a similar possibility for the distortion factor α . In fact, the latter can affect the perceived quality of the compressed image in a radical way.

In many cases it is desirable to keep a certain shape of the fovea. For instance, in images where foveae are located on human faces it is desirable to have it round or oval so that it better relates to the shape of the original object, that is, of a human face.

If the distortion factor is fixed, the shape of a fovea is dependent entirely on its absolute location in the image and its position relative to other foveae. This happens because scaling factors ensure a fixed compression ratio for the entire image and sample the pixels of the original image accordingly, not accounting for how dense or sparse sampling should be around a fovea.

If, however, we allow for variable distortion factors we can put more control on such sampling. Knowing the entire span that has to be sampled, as well as the region that has to be sampled with maximum detail, we can manipulate both scaling and distortion factors to achieve an optimal effect.

The basic formula for transforming the pixels of the compressed image to the ones in the uncompressed image is as follows¹:

$$d = \text{sgn}(v_d) \frac{\exp(\frac{|v_d|}{s}) - 1}{\alpha}. \quad (14)$$

d denotes the distance from a fovea either along the x or the y direction.

The proposed optimal distortion effect can be best described with the graph in Fig. 4. The curve in the graph represents (14) for fixed s and α . The point where the two lines intersect corresponds to the border point of the fovea, d_x . In fact, all pixels whose coordinates' absolute values are less than that of the border point will be sampled with no loss, whereas sampling of pixels beyond this border point will be sparse.

If instead of fixing α , we fix d_x , the boundary point, we can guarantee lossless sampling of the fovea. The problem is to find such α and s that the abscissa² of the crossing point

¹ d and other variables here may be applied to both x and y axes to produce d_x and d_y relatively.

² As a matter of fact the ordinate of this point equals its abscissa.

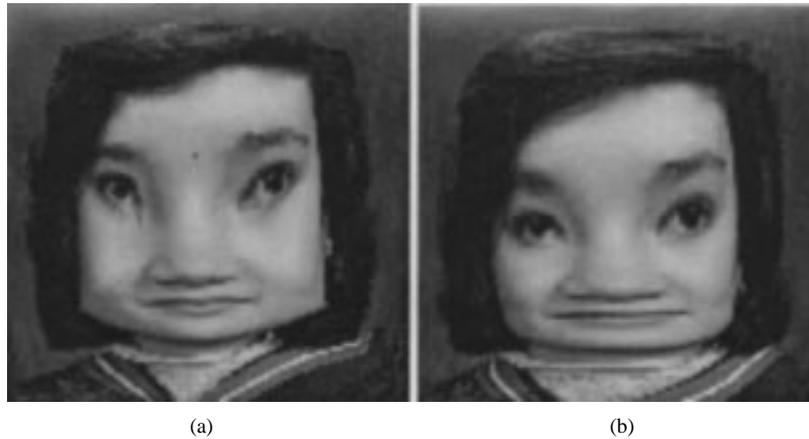


Fig. 2. Compressed image, fovea near center. Compression is 90%. (a) Modified and (b) Cartesian.



Fig. 3. Uncompressed image (CVR method).

of the two lines has the preset value of d_* and that (14) holds for $v_d = v_{d_{\max}}$ and for $d = d_{\max}$ for each of the coordinates of the image.

These two conditions, combined, result in the following system of equations:

$$\begin{aligned} |v_{d_{\max}}| \times \frac{D}{C_D} \times \left(\exp\left(\frac{|d_*|}{s}\right) - 1 \right) - |d_*| \\ \times \left(\exp\left(\frac{|v_{d_{\max}}|}{s}\right) - 1 \right) = 0 \end{aligned} \quad (15)$$

$$\alpha = \frac{\left(\exp\left(\frac{|d_*|}{s}\right) - 1 \right)}{|d_*|}$$

where D and C_D are dimensions of the original and compressed images, respectively. The first equation in (15) can be solved using numerical iteration. Assuming that $|d_*|$ is always less than $|v_{d_{\max}}|$, we can show that the function corresponding to the left part of (15) is continuous at the zero crossing. A

graph of this function for preset d_* and $v_{d_{\max}}$ is shown in Fig. 5.

We can define an iteration step which will converge to the zero crossing

$$s_{i+1} = \frac{|v_{d_{\max}}|}{\ln\left(\frac{v_{d_{\max}} \times D}{d_* \times C_D} \left(\exp\left(\frac{|d_*|}{s_i}\right) - 1\right) + 1\right)}. \quad (16)$$

Now, for every angle relative to a fovea we have to provide $v_{d_{\max}}$ and d_* , and use the iteration procedure to numerically compute s and α .

For example, if the shape of the fovea is a circle of radius r , then for every direction γ from the center of the fovea, the following formulae can be used to compute d_* :

$$\begin{aligned} d_{x_*}(\gamma) &= r \cos(\gamma) \\ d_{y_*}(\gamma) &= r \sin(\gamma). \end{aligned} \quad (17)$$

If, however, we allow for a variety of shapes, we can regard these different types of regions as fovea classes having their own methods for calculating d_* .

An example of a circular fovea (right) as opposed to a conformable fovea (left) is shown in Fig. 6. Here, the unsampled pixels are left black while normally their gray levels would be approximated based on those of their closest neighbors or using some other interpolation technique.

IV. MOVING FOVEA

When used to compress several images in a continuous sequence, the position of the foveae need not be the same for each image. If the positions do differ, when the images are viewed in sequence, the fovea will appear to move and may be called a *moving fovea*. One obvious place such a fovea would be useful is in a videophone application. The location of a fovea could follow a person's mouth, keeping that part of the image most clear. As the individual moves through the scene, so too would the fovea. The system must respond to the changes in the location of the fovea in real-time. Figs. 7 and 8 depict the effect of movement of the fovea. For the fovea location at F2, the results of the transform of only some of the points lying in the first quadrant can be obtained from the look-up table (LUT) when the fovea is at the center (F1). In this

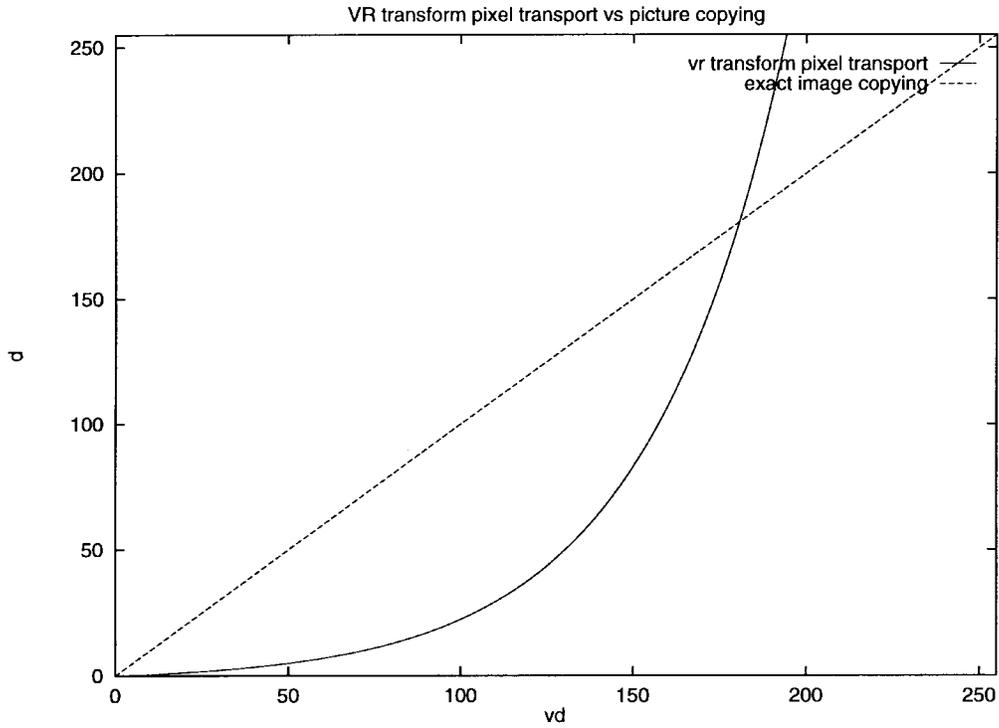


Fig. 4. Demonstration of VR transform versus exact image copying.

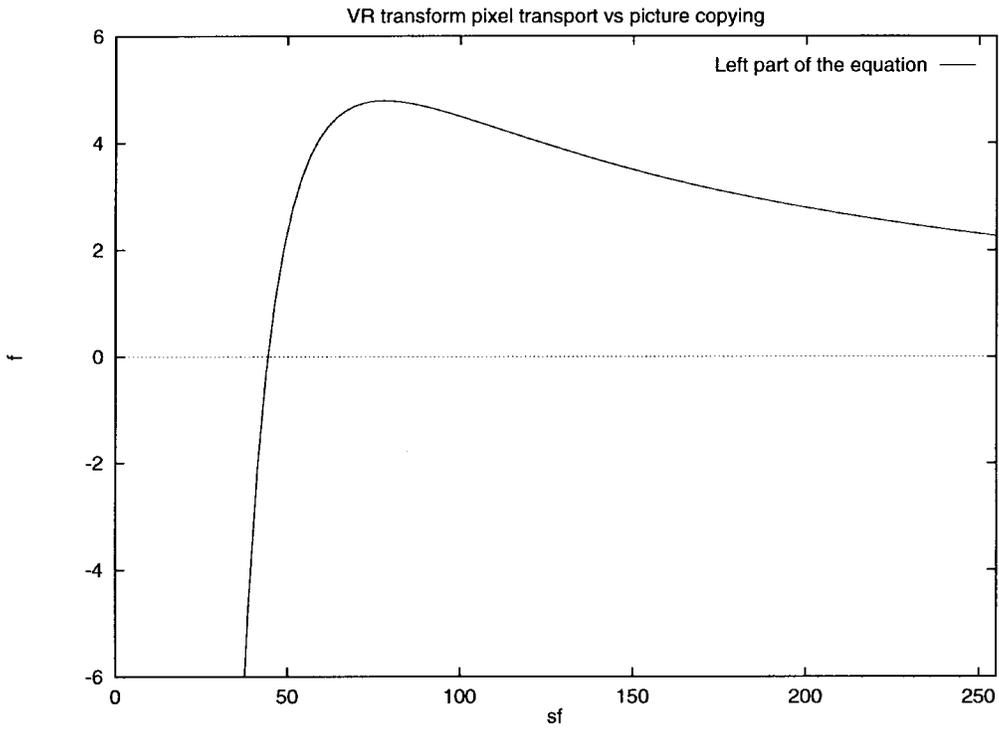


Fig. 5. Left part of the first equation in the system.

case we have a problem with the other points in that quadrant. Thus the best structure for the LUT is one that contains results of the transform for all points lying in the first quadrant, when the fovea is at the bottom left corner of the image. In this case the number of entries in the LUT is equal to the size of the image. For any location of the fovea, the relative coordinates

of the pixels in the image are obtained. Then the transform of the points $(x_{max}, 0)$ and $(0, y_{max})$ is computed to obtain the dimension of the VR image bounding box. Using this information the LUT is searched and the required information for performing the VR transform is obtained. Note that the VR transform for other quadrants can be related to the LUT

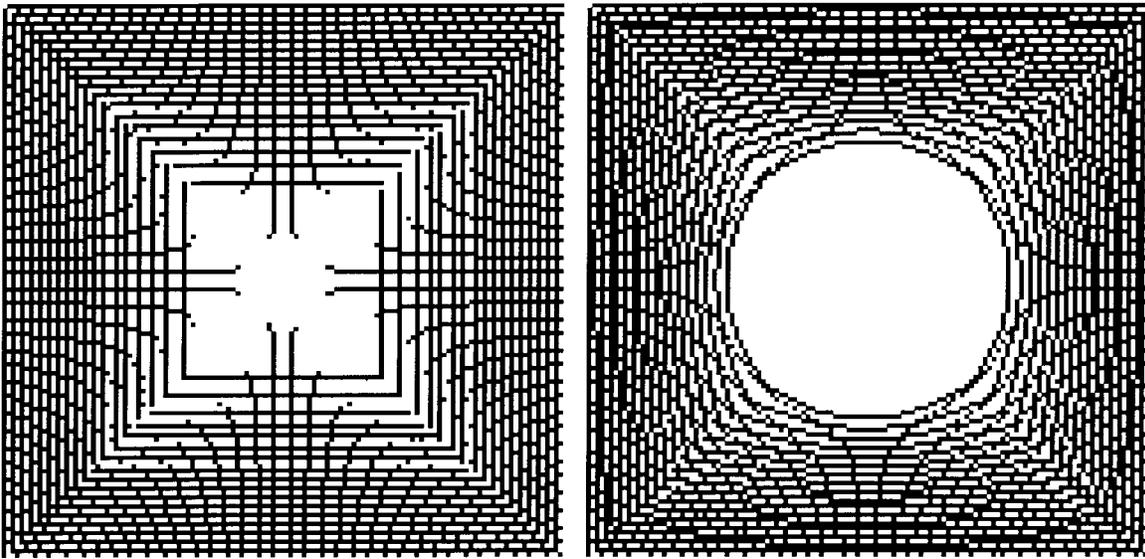


Fig. 6. Fixed distortion and conformable fovea versus variable distortion and circular fovea. A circular fovea is shown at right as opposed to a conformable fovea on the left.

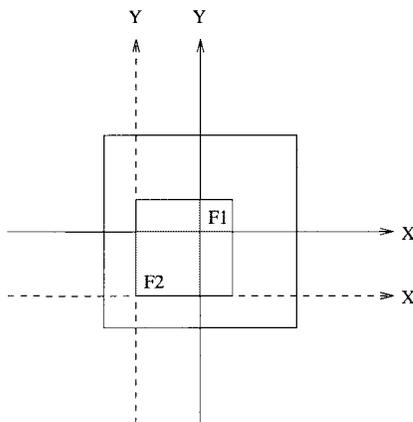


Fig. 7. Movement of the fovea.

for the first quadrant, thus it is sufficient to store information only for the first quadrant.

V. MULTIPLE FOVEAE

There may be situations where there is more than one area of interest to the observer. Such cases require multiple foveae, where two or more regions are displayed with higher resolution than the remainder of the image.

When additional centers of attention are introduced, a decision must be made to either reduce the resolution around each fovea to compensate, or retain additional information for each additional fovea, reducing the compression ratio. The quality of an image then depends on the relative position of multiple moving foveae.

There is no unique way of defining multiple foveae. However, one can clearly define the relationship depending on the desired properties. Two distinct approaches are *cooperative* and *competitive* foveae.

A. Cooperative Fovea

A method which works well for two foveae calculates the location of a point in the transformed image with respect to each fovea separately. The true location is then found by weighting the two estimated points according to the distance of the original point from the fovea. A higher weight is given to the location calculated using the closer fovea

$$l_{\text{actual}} = l_1 * \frac{d_2}{d_1 + d_2} + l_2 * \frac{d_1}{d_1 + d_2} \quad (18)$$

where l_i represents the coordinates of a point calculated using fovea i , and d_i represents the distance to fovea i .

This scheme is not limited to two foveae; a greater number of foveae simply involves computing a weighted average of a larger number of contributing points. The method is termed *cooperative* because all foveae contribute to the calculation of the position of a point in the transformed image.

A unique property of cooperative foveae is the existence of “visual streaks” between them. The area of highest quality³ in the scene will not only be at the foveae, but also along the line connecting the foveae. Multiple foveae systems will also contain areas of high quality between foveae, whether spots or curves.

The quality of areas outside of all foveae depends only on the proximity to the foveae. Quality is independent of relative positions of the foveae in that as long as the weighted average of all the foveae remains the same, the positions of the foveae do not make a difference to periphery quality. In a scene where multiple moving cooperative foveae exist, although “visual streaks” may appear between the foveae, the quality in the periphery will remain unchanged.

³An area is said to be of higher quality if it retains a larger ratio of pixels from the original scene.



Fig. 8. Examples of images with a moving fovea (approximately 91% compression.)



Fig. 9. Sample image before compression.



Fig. 10. Cooperative foveae placed on the outer faces. Compression is 80%.

B. Competitive Foveae

The definition of *competitive* foveae comes from the fact that all foveae compete to calculate the position of a point in the transformed image. The fovea which is closest to any point in the original image will be the one that determines its transformed position

$$l_{\text{actual}} = \{l_i : \forall \text{ foveae } j, j \neq i, d_i < d_j\} \quad (19)$$

where l_i represents the coordinates of a point calculated using fovea i , and d_i represents the distance to fovea i .

The “visual streaks” do not appear between competitive foveae, as the image is essentially broken up into separate regions, each having only one fovea contributing to the compression. There is no noticeable transition between regions, as the boundary is equidistant from the contributing foveae.

Here, the quality of an image depends not only on the proximity to a fovea, but also on the relative position of all the foveae. If the total compression ratio is set to a given constant, then the overall quality of the image will increase as two foveae move closer. This effect is more pronounced in the periphery of the scene. If two foveae are centered on the



Fig. 11. Competitive foveae placed on the outer faces. Compression is 80%.

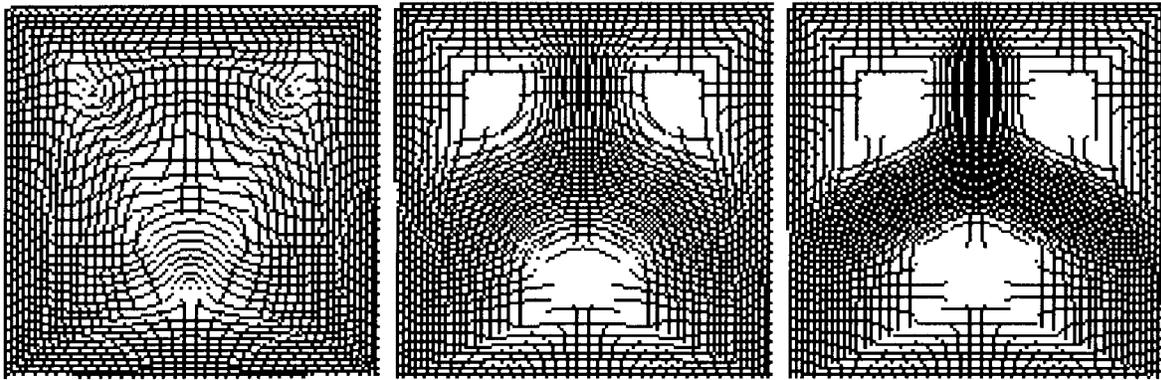


Fig. 12. $Power = 1$ (left), 4 (center), and 10 (right).



Fig. 13. Original image.

same location, then they act exactly as a single fovea would in that position. Figs. 9–11 show the results obtained using Cooperative and Competitive foveae on a given image.

C. Generalized Multiple Fovea

The concept of cooperative foveae can be generalized by creating weights depending on a certain *power* of distances to foveae. Varying *power* produces interesting effects on how cooperation among multiple foveae in the image takes place.

The general formula for calculating l_{actual} is as follows:

$$l_{\text{actual}} = \frac{1}{\sum_{j=0}^{N-1} \frac{1}{d_j^{\text{power}}}} \times \sum_{i=0}^{N-1} \frac{l_i}{d_i^{\text{power}}} \quad (20)$$

where N is the number of foveae, and *power* is the power with which weights corresponding to distances d_i are applied to l_i . If $N = 2$ and $power = 1$ then (20) reduces to (18).

It is clear that the ghost foveae disappear as the parameter grows, and the quality of the image in the foveae improves

at the expense of quality between the foveae. If the *power* is large, the overall effect of compression is similar to that achieved when competitive foveae are used. The results of applying different *power* parameters are shown in Fig. 12.

1) *Combining Varying Distortion and Scaling Factors with Multiple Foveae*: Multiple foveae can be combined with the concept of optimal variation of distortion and scaling factors (Section III-C). Fig. 13 shows an original image before compression. Figs. 14 and 15 show the advantage of the modified approach over the conventional competitive foveae. Notice that the faces are clearer in the image for competitive foveae with variable distortion. We used five foveae in this example.

VI. ENHANCING JPEG

Spatially varying sensing can be used very effectively in enhancing the performance of JPEG at high compression ratios. Fig. 16 shows the original image without compression. This image is of size 512×512 , i.e., 262 184 bytes. The image



Fig. 14. Competitive with variable distortion.



Fig. 15. Competitive without variable distortion.

after compression using JPEG with quality = 1 is shown in Fig. 17, which has 3798 bytes of data (98.55% compression) and signal to noise ratio (SNR) root mean square (rms) of 18.25.

The result of JPEG enhanced by a VR preprocessor is shown in Fig. 18. The image is compressed to 3776 bytes (98.56%) compression and the SNR (rms) is 47.23. The difference in quality is probably much more noticeable to our perception than can be explained by these numbers. An alternate way of combining VR with JPEG would be to vary the quality parameter (Q) in JPEG compression depending on the distances from the foveae.

At present we are working on enhancements to wavelet and MPEG compression methods. We will also investigate the applications of automatic selection of points/regions of interest to enhance compression methods. First, automatic detection of facial features will be used to improve compression in mug-shot databases; then the problem will be addressed for arbitrary scenes.

VII. A VIDEOCONFERENCING SYSTEM

Based on the experience with VR compression, a prototype of a multimedia system is being built. Motion video with audio (videophone/teleconferencing), a mug shot database, a



Fig. 16. Original image.



Fig. 18. Image with VR/JPEG hybrid compression.



Fig. 17. Image with JPEG compression.

collaborative drawing component, and an editing component are being integrated. The system can function in a generic environment with little or no additional hardware. Such a system is inexpensive, easy to upgrade and maintain, and portable across many systems.

The videophone component is able to provide transmission of grey scale images from an image *server* to a *display* or *viewer* process. The server process is responsible for capturing, compressing and transmitting the image. The display process accepts images, uncompresses them, and displays them on a screen. Fig. 19 shows a sample videoconferencing window.

In addition to variable resolution, additional compression is provided by an intraframe difference encoding routine. The difference between pixels in successive frames is found (most pixels will not change if the image is static) and only changed pixels are transmitted.

Currently, frame rates around ten to 12 frames/s have been achieved on SGI workstations across an ethernet. With the compression values obtained (up to 98% with interframe encoding) the network can easily handle much higher frame rates. It must be noted that the system is based on software; thus, no special boards except digitizers are required.

The teleconferencing component operates on the same principle as the videophone, except that it does not have one fixed fovea at the center. Instead, multiple foveae can be placed at the user's discretion. Work is currently under way to implement moving foveae which automatically track any person or other moving object in a scene, as chosen by the observer.

VIII. CONCLUSION

Variable resolution has some major advantages as a teleconferencing technique. The rate at which it can compress images, especially on machines with limited processing speed, and the high quality present in the foveal region, makes it ideal for the multimedia market. Our experience with the multimedia prototype supports this belief. Most importantly, frame rates can be maintained without the need for additional hardware. Using this approach an organization can implement a teleconferencing system on a local or wide area network with very little hardware cost. Our study has demonstrated the advantage of VR in enhancing the performance of the JPEG algorithm at high compression ratios.

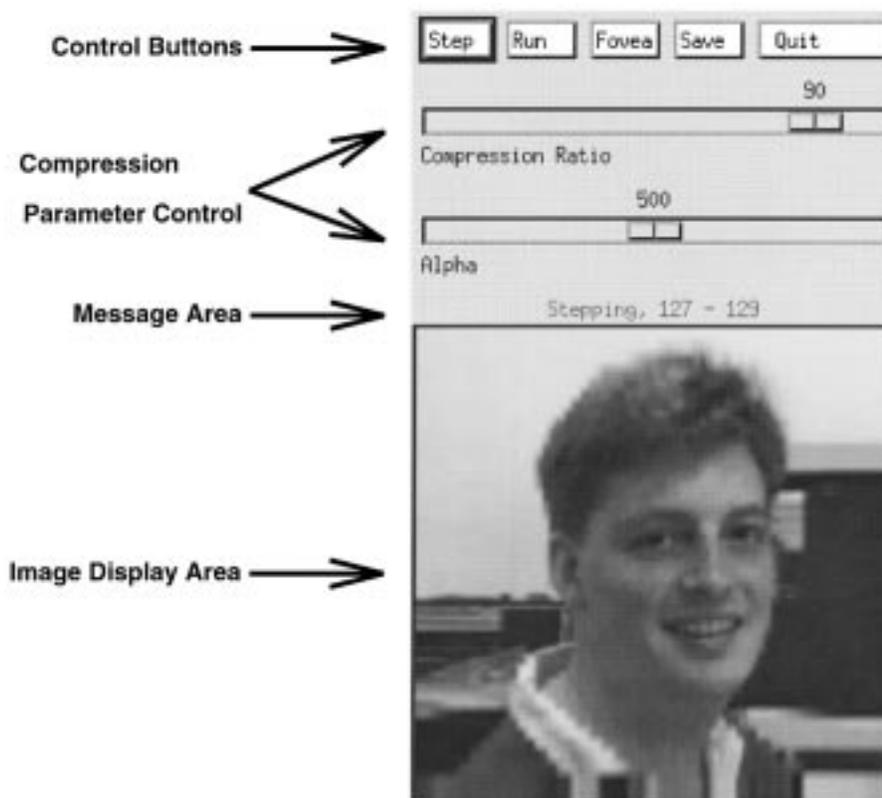


Fig. 19. Sample videophone window.

At present we are working on demonstrating how our models for multiple foveae can be used to approximate various animate visual systems. In future work we will study techniques for transmission of VR images over ATM networks.

ACKNOWLEDGMENT

The assistance of K. Richine in some of the experiments is greatly appreciated.

REFERENCES

- [1] D. Anastassiou, M. Brown, H. Jones, J. Mitchell, W. Pennebaker, and K. Pennington, "Series/1 based videoconferencing system," *IBM Syst. J.*, vol. 22, nos. 1/2, pp. 97–110, 1983.
- [2] F. Argenti, G. Benelli, and C. Kutufa, "Transmission of wavelet transform coded TV images on a noisy channel," in *IEEE Int. Conf. Communication*, 1993, pp. 386–390.
- [3] F. Argenti, G. Benelli, and A. Mecocci, "Source coding and transmission of HDTV images compressed with the wavelet transform," *IEEE J. Select. Areas Commun.*, vol. 11, pp. 46–58, Jan. 1993.
- [4] A. Basu and S. Licardie, "Modeling fish-eye lenses," in *IEEE IROS Conf.*, Yokohama, Japan, July 1993, pp. 1822–1828.
- [5] A. Borning and M. Travers, in *Two Approaches to Casual Interaction Over Computer and Video Networks CHI'91 Conf. Proc.*, S. Robertson, G. Olson, and J. Olson, Eds., ACM, pp. 13–19.
- [6] A. Califano, R. Kjeldsen, and R. M. Bolle, "Data and model driven foveation," in *IEEE Int. Conf. Pattern Recognition*, Atlantic City, NJ, June 1990, pp. 1–7.
- [7] M.-S. Chen, Z.-Y. Shae, D. Kandlur, T. Barzilai, and H. Vin, "A multimedia desktop collaboration system," in *IEEE Globecom'92*, 1992, pp. 739–746.
- [8] M. Antonini *et al.*, "Image coding using wavelet transform," *IEEE Trans. Image Processing*, vol. 1, pp. 205–220, 1992.
- [9] O. Inzunza *et al.*, "Topography and morphology of retinal ganglion cells in *falconiforms*: A study on predatory and carrion-eating birds," *Anatomical Rec.*, vol. 229, pp. 271–277, 1991.
- [10] P. H. Westerink *et al.*, "Subband coding of images using vector quantization," *IEEE J. Select. Areas Commun.*, vol. 6, pp. 713–719, June 1988.
- [11] Y. Zhang *et al.*, "Variable bit rate video transmission in B-ISDN environment," *Proc. IEEE*, vol. 79, pp. 214–221, Feb. 1991.
- [12] D. Le Gall, "MPEG: A video compression standard for multimedia applications," *Commun. ACM*, vol. 34, pp. 46–58, Apr. 1991.
- [13] B. P. Hayes and M. de L. Brooke, "Retinal ganglion cell distribution and behavior in procellariiform seabirds," *Vision Res.*, vol. 30, no. 9, pp. 1277–1289, 1990.
- [14] A. Hughes, "The topography of vision in mammals of contrasting life style: Comparative optics and retinal organization," in *Handbook of Sensory Physiology: The Visual System of Vertebrates*. Berlin, Germany: Springer-Verlag, 1977, vol. VII/5, pp. 697–756.
- [15] A. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Trans. Image Processing*, vol. 1, pp. 18–30, Jan. 1992.
- [16] S. S. Easter Jr., "Retinal growth in foveated teleosts: Nasotemporal asymmetry keeps the fovea in temporal retina," *J. Neurosci.*, vol. 12, pp. 2381–2392, June 1992.
- [17] P. Kirstein and J. Crowcroft, "International links for research collaboration," *Comput. Networks ISDN Syst.*, vol. 21, pp. 261–273, 1991.
- [18] W. Li and M. Kunt, "Block adaptive 3-D subband coding of image sequences," in *IEEE Int. Conf. Communication*, 1993, pp. 532–536.
- [19] M. Liou, "Overview of the p*64 kbit/s video coding standard," *Commun. ACM*, vol. 34, pp. 59–63, Apr. 1991.
- [20] J. N. Lythgoe, *The Ecology of Vision*. Oxford, U.K.: Oxford Univ. Press, 1979.
- [21] B. Müller and L. Peichl, "Horizontal cells in the cone-dominated tree shrew retina: Morphology, photoreceptor contacts, and topographical distribution," *J. Neurosci.*, vol. 13, pp. 3628–3646, Aug. 1993.
- [22] M. Nelson, *The Data Compression Book*. M & T Books, 1992.
- [23] S. Panda-Jonas, J. B. Jonas, M. Jakobczyk, and U. Schneider, "Retinal photoreceptor count, retinal surface area, and optic disk size in normal human eyes," *Ophthalmology*, vol. 101, no. 3, pp. 519–523, 1994.
- [24] J. S. Pointer, "The cortical magnification factor and photopic vision," *Biol. Rev. Cambridge Philosoph. Soc.*, vol. 61, pp. 97–119, 1986.
- [25] E. Reusens and T. Ebrahimi, "New results in subband/wavelet image coding," in *IEEE Int. Conf. Communication*, 1993, pp. 381–385.
- [26] G. Sandini and V. Tagliasco, "An anthropomorphic retina-like structure

- for scene analysis," *Comput. Graph. Image Process.*, vol. 14, pp. 365–372, 1980.
- [27] G. Sandini and M. Tistarelli, "Vision and space-variant sensing," in *ECCV-94 Workshop Natural Artificial Visual Sensors*, Osqulda's vag 6, Stockholm, Sweden, May 1994, pp. 398–425.
- [28] E. Schwartz, "Computational anatomy and functional architecture of striate cortex: A spatial mapping approach to perceptual coding," *Vision Res.*, vol. 30, pp. 645–669, 1980.
- [29] P. Sen, B. Maglaris, N. Rikli, and D. Anastassiou, "Models for packet switching of variable-bit-rate video sources," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 865–869, June 1989.
- [30] T. Senoo and B. Girod, "Vector quantization for entropy coding of image subbands," *IEEE Trans. Image Processing*, vol. 1, pp. 526–532, 1992.
- [31] S. Sinclair, *How Animals See*. Washington, DC: Library of Congress, 1985.
- [32] Á. Széll *et al.*, "Different patterns of retinal cone topography in two genera of rodents," *Mus and Apodemus, Cell and Tissue Research*. Berlin, Germany: Springer-Verlag, 1994, pp. 143–150.
- [33] M. Tistarelli and G. Sandini, "Dynamic aspects of active vision," *Comput. Graph. Image Process.*, vol. 14, pp. 365–372, 1992.
- [34] F. Tong and Z. N. Li, "The reciprocal-wedge transform for space-variant sensing," in *IEEE Int. Conf. Computer Vision*, 1993, pp. 330–334.
- [35] ———, "Reciprocal-wedge transform in motion stereo," *IEEE Int. Conf. Robotics Automation*, 1994, pp. 1060–1065.
- [36] G. Vecchietti, M. Modena, and G. Parladori, "ATM network and VBR video source: Data structure and statistical analysis of an implemented HYBRID DCT broadcast video encoder," in *GLOBECOM*, pp. 34–39, 1991.
- [37] W. Verbiest and L. Pinnoo, "A variable bit rate video codec for ATM networks," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 761–770, June 1989.
- [38] G. Wallace, "The JPEG still picture compression standard," *Commun. ACM*, vol. 34, pp. 31–44, Apr. 1991.
- [39] T. Welch, "A technique for high performance data compression," *IEEE Comput.*, vol. 17, pp. 645–669, June 1984.
- [40] S. Zafar, Y. Q. Zhang, and B. Jabbari, "Multiscale video representation using multiresolution motion compensation and wavelet decomposition," *IEEE J. Select. Areas Commun.*, vol. 11, pp. 24–35, Jan. 1993.



Anup Basu (S'89–M'90) received the B.S. degree in mathematics and statistics and the M.S. degree in computer science, both from the Indian Statistical Institute. He received the Ph.D. degree in computer science from the University of Maryland, College Park.

He has been with Tata Consultancy Services, India, and Strong Memorial Hospital, Biostatistics Division, Rochester, NY. At present, he is an Associate Professor, Department of Computer Science, University of Alberta, Edmonton, Alta., Canada. His

research interests include computer vision, robotics, and multimedia.



Kevin James Wiebe was born in Vancouver, B.C., Canada in 1969. He received the B.Sc. in computer science and mathematics in 1992 from Simon Fraser University, Burnaby, B.C. He received the Ph.D. degree in computer vision from the University of Alberta, Edmonton, Alta., Canada, in 1996. His Ph.D. thesis involved novel image/video compression and encoding techniques motivated by unique characteristics of animate visual systems.

His interests lie in multimedia technologies, image/video processing and compression, anthropomorphic computer vision, and expert systems. He has worked for Sony Corp., Tokyo, Japan, and IBM Canada, Toronto, Ont., researching topics in hypermedia systems, databases, and software development. He is currently involved with his own software company, focusing on advanced GIS knowledge extraction and manipulation research.