MID-LEVEL VISION: NEW DIRECTIONS IN VISION AND VIDEO

Edward H. Adelson†, John Y. A. Wang‡, and Sourabh A. Niyogi¶

†Department of Brain and Cognitive Sciences
‡Department of Electrical Engineering and Computer Science
The MIT Media Laboratory, Massachusetts Institute of Technology,
Cambridge, MA 02139

ABSTRACT

Human vision, machine vision, and image coding, all have to deal with the problem of finding representations that are useful and efficient. The best-known techniques today are based on low-level processing, using signal processing concepts such as filters, transforms, and simple non-linearities. Low-level concepts are at the heart of standard vision systems for computing optic flow, texture, etc. Low-level image coding techniques include DCT’s, pyramids, wavelets, etc. To advance to a new generation of image coding architectures we need to work with new image representations that involve such concepts as surfaces, lighting, transparency, etc. These representations fall in the domain of “mid-level” vision. By representing images with these more sophisticated vocabularies we can increase the flexibility and efficiency of our vision and image coding systems. In one application, we decompose an image sequence into a set of overlapping layers, rather like the “cels” used by a traditional animator. These layers are ordered in depth, sliding over one another and being combined according to the rules of transparency and occlusion. For some test sequences we achieve data compression far better than is possible with standard techniques such as MPEG.

1. INTRODUCTION

Human vision, computer vision, and image coding all must deal with some common problems. In each case, it is necessary to represent images and image sequences in a manner that is efficient and useful. One may also speak of a hierarchy of representations, from low-level to high-level, for each domain.

In the case of vision, low-level operations involve the analysis of local motion, color, orientation, and spatial scale, using operations such as linear filtering and simple non-linearities. In image coding we find similar low-level operators being used in the form of linear transforms, which extract information about local orientation, scale, and color, as exemplified in the DCT or in various pyramid decompositions. Low-level vision and low-level coding are well understood today and they work.

At the other extreme are high-level systems. In vision, this involves recognizing and understanding the objects and actions taking place within a scene. A high-level image coding system would rely on computer vision to analyze the scene into a rather abstract description – e.g., “A red Chevrolet is chasing a black BMW down a dirt road,” along with various details. These high-level descriptions then provide sufficient information for computer graphics systems to render and synthesize an image matching the description. High-level image coding offers the possibility of extremely high compression, and can place the image information in a much more useful form than mere pixels or transform coefficients.

The problem with high-level image coding is that it depends on solutions to difficult machine vision problems that have no prospect of a general solution in the near future. There are two strategies one can take: (1) Find high-level solutions in restricted image domains, and (2) find mid-level solutions in more general image domains.

The restricted domain approach has shown promise in the case of coding talking head images for teleconferencing. Typically, a 3-D wire-mesh model is fit to the face data and the parameters needed to animate the model are transmitted, along with a single texture map of the face itself [2]. A wire-mesh of a talking head does not do much good when one is trying to transmit a car chase, or a tennis game. To transmit these scenes requires either a larger dictionary of 3-D models, or an approach to image coding that is more easily generalized.
Figure 1: (a) An image of two overlapping transparent squares. (b) Naive decomposition of the image. (c) Perceptual decomposition of the image.

Note that low-level systems can get along with a very simple fixed vocabulary; in the case of a DCT the vocabulary consists of transform coefficients. Any scene, no matter how complex or unusual, can be described within this vocabulary. For a high-level system the vocabulary must become far richer to accommodate the complexity of the world.

It seems that low-level coding is now approaching a plateau in performance. Further work may allow small improvements in compression ratio or speed, but there will be no major leaps forward. At the same time, high-level coding is out of reach except in restricted domains.

We believe that the most promising area for research today is mid-level vision and mid-level image coding. The vocabulary of mid-level representations includes surfaces, lighting, segmentation, global motion, texture, and so on.

2. MID-LEVEL VISION CONCEPTS

Our own work has recently concentrated on decomposing an image sequence into a set of overlapping 2-D layers, in order to describe images in terms of a visual vocabulary similar to that used in traditional "cel" animation as well as computer graphics compositing. The basic representation is inspired by studies in the perception of transparency. Consider figure 1(a). If we were to segment it using standard image processing techniques, we would end up with something like that shown in figure 1(b), where the segments are based on gray-level. However, we humans look at it and segment it perceptually into the two overlapping squares shown in figure 1.

The information in a given layer combines with those of the layers beneath it to generate an image. The basic ingredients of a layered representation are these: (1) an intensity map indicating the luminance (and/or color) that is added by the layer at each point; (2) an attenuation map, also known as an alpha channel, indicating a multiplier between 0 and 1 that is applied to the layers beneath the current one; (3) a warp map indicating how this layer is to move and distort over time.

If one can generate such a description from a scene then the synthesis of the original scene is easy. The hard part is the analysis—getting the scene into the layered representation in the first place. We have devised some techniques that can be applied to moving image sequences in order to decompose them into layers [1, 4].

3. MOTION LAYERS OF 3D PLANAR SURFACES

Some sequences lend themselves naturally to a layered analysis. For example, the MPEG flower garden sequence consists of a rigid scene viewed by a camera out of a passing car window; figure 2(a) shows a single frame. The various parts of the scene can be considered to slide past one another while undergoing simple distortions. We can model the distortions as affine transformations. This model is imperfect but it is quite adequate for the purposes of decomposing this sequence.

In our motion decomposition, the criterion for assigning pixels into a layer depends on the particular motion of the pixel. For example, by grouping all pixels undergoing a similar affine motion, we can identify the points in the image belonging to single planar surface.

Motion layer extraction is a difficult problem requiring simultaneous estimation of motion, surface intensity map, and alpha map. We have developed iterative techniques based on robust statistics that identify the affine motion regions and extract the motion layers. Our current implementation deals only with occlusion surfaces where the alpha maps take on values of 1 and 0. However, we find that this simple model is adequate in representing many natural scenes.

Our approach to segmentation relies on a gradual migration from low-level pixel motion description to mid-level surface affine motion description. Pixel motion is estimated with a coarse-to-fine gradient motion estimator. From pixel motion we formulate a set of likely affine motion models. Affine regions are determined by assigning each pixel to the model that best describes the motion of the pixel. In addition, we can determine the occlusion relationships of the each of the layers with respect to the other layers. These layer maps can be further compressed by low-level vision techniques such as transform coding. Thus, embedded in our motion layers is a hierarchy of complexity.
ranging from low-level vision to mid-level vision concepts.

Figure 2(a) shows one frame of the MPEG sequence. Figure 2(b) shows the layered representation of the sequence consisting of 3 primary layers. In this example, the complexity of the 3D scene is reduced to 3 planar surfaces: the tree layer, the house layer, and the flowerbed layer. The placements of these layers in this figure indicate the respective depth of the layers. Given these layer intensity maps and their associated motion maps, we can synthesize the original sequence. We have achieved good image quality at rates of 300-600 kbits/sec, which is substantially better than can be achieved by standard MPEG coders [5]. One frame of that sequence is shown in figure 3(a).

Because the layered description relies on mid-level vision concepts, we can perform useful image manipulation on the data in ways that are not possible with low-level vision systems. For example, we can choose to omit the tree when synthesizing the sequence. One frame of this edited sequence is shown in figure 3(b).

In addition to data compression, our motion layers also provide solutions to problems in motion estimation. The layers correctly represent the motion at an occlusion boundary. Motion estimators relying on low-level concepts produce only one motion at each point in the image and have difficulties in dealing with motion discontinuities at these occlusion boundaries. The layered description easily deals with this problem by producing multiple motions in these regions and by using the alpha maps to explain the discontinuities in motion. Likewise, physically, discontinuities occur from the change in opacity (alpha map) of the different surfaces.

4. NONRIGID DEFORMABLE LAYERS

Another problem is generalizing the layer concept to non-rigid objects. We have investigated one particular case, namely a person walking in the frontoparallel plane. At present we simplify the problem by ignoring the arms and modeling the walker as consisting of two layers, as shown in figure 4. One layer contains the upper body plus one leg; the other contains the upper body plus the other leg. These layers are flexible and can distort in order to follow the observed motion.

We analyze a video sequence into this representation by considering it as a three-dimensional volume in XYT. A person walking through this space traces out a characteristic 3-D pattern. We fit a deformable surface to this pattern in order to capture the motion [3]. A single frame of a walking person is shown in figure 5(a), and the corresponding surface is shown in figure 5(b).

At present our work with walkers is limited to the analysis side, but we will extend it to the problem of synthesis in the future. In principle it should be possible to code a walking sequence with very few bits.

5. CONCLUSIONS

Future image coding systems can be more efficient and more flexible if they are based on mid-level vision concepts. The concept of layers is already a basic part of the visual vocabulary used in computer graphics and animation. Layers are overlapping 2-D objects that incorporate several registered maps, including information about intensity, transparency, and motion. We propose that such a representation should be a central part of future image coding systems. The difficult remaining challenge is achieving a layered decomposition given an image sequence. We are developing a set of techniques for that purpose, which we have applied to a few specific sequences. More work on mid-level vision is needed to make this into a general purpose scheme.

6. REFERENCES


Figure 2: (a) A single frame of original MPEG sequence. (b) Layered decomposition of the sequence.

Figure 3: (a) A single frame of reconstructed sequence from layers. (b) Reconstructed image with tree omitted.

Figure 4: Layered representation of a walker.
Figure 5: (a) Single frame of walking sequence showing deformable model fit. (b) Recovered spatiotemporal surface of walker.