Perceptual Significance Hierarchy: A Computer Vision Theory for Color Separation[1]

Deborah Walters and Ganapathy Krishnan

Computer Science Department, University at Buffalo (SUNY), Buffalo, NY

Abstract

A Perceptual Significance Hierarchy (PSH) for line art images is developed which represents the relative perceptual significance of each image component. This is possible through the use of a set of image-features which are used by the human visual system. The PSH and related rho-space computer vision algorithms can be used to automate the fake color separation process used by the printing industry. This is accomplished by adding rudimentary visual processing capabilities to a computer graphics system.

I. Introduction

This paper describes an application of Artificial Intelligence techniques to a pre-press problem in the printing industry, color separation. This application area is interesting as it is one where expert systems techniques are not useful, where rule-based reasoning is inappropriate, and where relational knowledge bases make no sense. Instead, AI techniques based on basic visual perceptual computations and the parallel processing of visual information are required. This can be accomplished through the use of a Perceptual Significance Hierarchy (PSH), as described below.

A. The Color Separation Application

The printing industry is rapidly becoming an entirely electronic computer-based industry. However not all of the pre-press processes have been successfully computerized. Color separation is one such process. Before a colored image can be printed, it must go through the color separation process, which creates three or four separate plates - one each for printing cyan, magenta, yellow, and if required, black.

There are two types of color separation: process and fake. Process color separation is used for photographic images, and has been successfully automated by using colored filters to optically separate the colors projected from the negative of a color photograph. An example of images printed from process color separation are the color photographs in weekly news magazines. Fake color separation is used for line-art images such as the Sunday comics or commercial art. In this case, the printing company receives black and white art-work, and information about the color for each image region. The task is then to color in, by number, each region in the image. Despite the simple concept, this is a difficult problem.

At present there are two techniques for fake color separation: one is completely manual; and the other uses computer graphics techniques. In the manual technique, for each particular desired color in the image, a sheet of transparent acetate is laid on top of the original line drawing art, and blocks of translucent red cellophane tape are laid over each region to be colored the particular color. The red tape is then cut to the exact shape of the areas to be colored with an Xacto knife, and the excess tape peeled away. The acetate sheets are kept aligned with each other and with the original artwork by punching holes in their margins which fit over precisely aligned registration pins. This labor intensive manual color separation is still widely used.

In the latter technique the artwork is digitized and displayed on a computer graphics workstation. The user can specify a particular color using a color palette, and then can indicate which region should be filled with that color by pointing to the region with the cursor. A seed-fill algorithm can then be used to fill the desired region with the selected color [1]. After the user has interactively colored in all of the image, the graphics system can compute the three or four required printing plates.

There are several basic problems with the computer graphics based techniques, as illustrated by the simple line drawing in Figure 1:

1) First, extra lines in a region can cause problems. For example, if the bottom of the container was to be colored blue, then the user would have to separately select and fill each of the

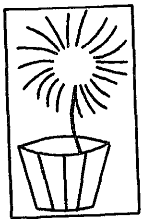

Figure 1

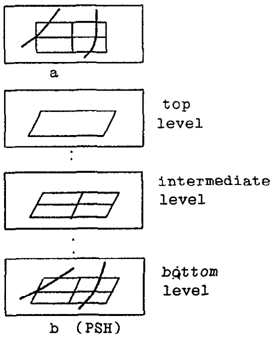

Figure 2

---

[1] This work supported in part by National Science Foundation grant IST-8409827 awarded to the senior author.

four regions of the bottom of the container. The human visual system can readily segment the entire bottom of the container into a single region, and it would be preferable if the user could fill such a "perceptual region" with a single fill command. For this particular example, the added cost of having to fill each image region may not be too large, but in many images this problem is more severe as artists often use lines to indicate texture, patterns, and interior lines, and these lines can divide a single perceptual region into many separate regions. Even in complex cases the user could point separately to each small region, but quite rightly, they refuse to use such an inefficient algorithm.

2) This problem of subdivided regions can also arise when lines from separate objects intersect. In Figure 1 the stem of the flower intersects the top boundary of the container, and divides the container top into two regions, which would again require separate fill commands.

3) Third, many regions are not surrounded by a closed contour. For example, there is a gap in the boundary of the top of the container. Such gaps may not seem significant to human observers, they understand that the boundary encloses a single region. But such gaps create problems for seed-fill algorithms as the color assigned to a region will "leak" out through any gaps and fill the surrounding region.

4) The fourth problem is that some regions in line drawings are defined, not by lines, but by illusory contours, as in the center of the flower in Figure 1. Again, in such situations seed-fill algorithms are useless.

There are two basic solutions currently used by the printing industry to solve these problems. The first solution is to simply make restrictions on the type of artwork that can be separated using the computer graphics techniques. One potential means would be to require that all artists and cartoonists that submit line drawings for color separation ensure that none of the four problems would arise in their work. This approach is obviously not possible as it would dramatically interfere with artistic license. Some pre-press companies do require that artwork be generated on a computer graphics workstation using techniques which do not allow images with any of the four problems to be produced. But this solution is only possible when the pre-press industry handles both the color separation and the image creation stages. All other types of images have to be separated using manual techniques.

Other pre-press companies handle these problems by manually retouching artwork before it is digitized. A technician will "white-out" the extra texture and pattern lines that divide single regions, and will draw in black connecting lines wherever there are gaps or illusory contours. The retouched art-work can then be digitized and colored in interactively. This is much more efficient than the completely manual technique, but is still very labor- intensive. In fact, the retouching stage requires as much time as the interactive coloring, which makes the system both expensive and slow.

B. Goal for AI Techniques

The goal of this research is to apply AI techniques to remove the retouching stage of color separation by enabling a computer to perform much of that preprocessing. Current electronic color separation systems represent a division of labor between human and machine, where a human visual system performs the segmentation of an image into objects or significant parts of objects to be colored - this is done in the retouching stage - and the machine takes care of most of the detail of filling in the segments with color and separating the colors into their primary components. The goal of this research was to give the machine a rudimentary visual sense, so that it can perform some, but not necessarily all of the segmentation process. Any portion of the retouching stage that could not be handled by the machine vision preprocessing system would be done by the user at the graphics workstation.

The goal was not to create a computer image understanding or scene analysis system, as the machine does not have to know what an object is, in order to color it, but only that a given region constitutes an object. The computer vision system just needs to be able to segment an image into regions which correspond to objects or object parts. And this can even be done with human interaction.

C. Perceptual Significance Hierarchy

One theoretical approach to this problem can be expressed in terms of a Perceptual Significance Hierarchy (PSH). There are various hierarchical image processing and image interpretation techniques. For example, many pyramid algorithms are based on the hierarchy of spatial resolution [2]. Another example would be hierarchical object representation, where an object is represented at various levels of detail [3]. In contrast, the PSH is based upon the perceptual significance of image components. Thus instead of representing an object at various levels of detail, the PSH would have a top level where only the most significant image components were represented, intermediate levels where the next most significant components would appear, and a bottom level where all image components would be represented.

For example, most observers say that the large rectangle is the most perceptually significant component of the scene in Fig. 2a, and that the unconnected diagonal lines are the least significant components. This could be represented in a PSH as shown in part b of Fig. 2, with the rectangle represented as being most significant by it's presence at the top level. Similarly, the interior lines of the rectangle are represented in the PSH as being of intermediate significance, and the diagonal lines as having the least perceptual significance.

A PSH would have several uses. First, if a PSH can be computed in parallel over an entire image (as is possible using the method described in Section III), then the PSH can be used to focus the attention of subsequent non-parallel techniques. For example, the PSH could be used to improve the efficiency of model-matching by having only the most perceptually significant image components matched first. Another use of a PSH could be to aid in segmentation (see Section IV).

The goal of the theoretical portion of this research is to develop computer vision algorithms which produce a PSH. The approach taken is to look to the human visual system for inspiration. This approach will not always be useful in computer vision as the solutions used by the human visual system may be based on limitations of the neural hardware which do not exist in computer systems. However, in other cases, the human solution may be based on the general problem of vision, and thus be useful for machine vision systems as well. Using the human visual system for inspiration is especially relevant for the color separation application as artists may use a semantics of line art that is based on the semantics used by humans in the visual perception of line art.

II. Human Determination of Perceptual Significance

How do humans perceive line art; what aspects of line drawings are perceptually significant? Psychophysical experiments have been used to provide answers to these questions [4]. The experimental results show that the local connections between the ends of lines and curves are important. In fact, before the action of context or other cognitive effects, the perceptual significance of a line or curve is based on the type of connection at it's two ends. There is actually a hierarchy of connection types as illustrated in Figure 3, where an instance of each of the three possible types of end connections are shown. Lines terminating in Type A connections are the most perceptually significant, those with B are the next most significant, and so on [5].

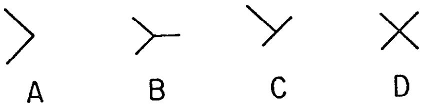

Figure 3

This hierarchy of line end connections contains some of the connections which have been previously used in computer vision algorithms: corners [6], and 'L', 'T', and 'fork' junctions [7-10]. But this end connection hierarchy is different in several ways: first, with the addition of one additional connection type it can be applied to both straight and curved lines; second, it is domain independent; third, only four types of connections are present; and fourth, this hierarchy has been proven to be geometrically complete, thus all potential connections between the end and interjacent points of lines can be classified as one of the four connection types A, B, C or D [5]. However, the most important difference is that these end connection features will not be used for constraint labeling as a part of object interpretation, but rather for a lower level task such as image segmentation.

These end connection features can be used to develop a PSH. But first it is interesting to ask "Why should the end connection features be significant?" What scene invariants do they capture? This can be explored by making two simple assumptions. First, assume that the viewpoint is representative [11,12]. This means we assume we are not looking at a scene from a limited number of viewpoints which cause the objects to appear to be accidentally aligned. The second assumption is that the objects are not accidentally aligned.

From these assumptions it is possible to make various inferences about the perceptual significance of the end-connection features [13]. For example, it can be shown that lines associated with the type A connection have a higher probability of having arisen from the bounding contour of an object, while the type B are more likely to have arisen from objects than non-objects. The type C connections are most likely to represent one object occluding another, and the type D connections would most likely arise from the intersection of two wires or transparent edges.

So the hypothesis is that these features are important because it is more probable that such features would arise from the physical properties of objects in a scene than from other causes. Thus the lines associated with these features should contain more information about objects in a scene than lines lacking these features, and thus be more perceptually significant.

III. End Connections and the PSH

We can use this hypothesis to develop a technique for selectively enhancing an image by categorizing lines and curves in terms of their end connections. The algorithm for producing the PSH assumes that the perceptual significance of a line is determined by the type of connection at each end of the line. This assumes that a line has exactly two ends, which is possible if lines are assumed to lack discontinuities in orientation, and if branching lines are assumed to be broken at branching intersections [14]. It will be further assumed that a single type A connection makes a line more perceptually significant than having two type B connections, and that a single type B connection implies more significance than a single type C connection. Thus a PSH can be created by assigning those lines which have type A connections at both ends to the top level of the hierarchy (which will be referred to as Level 1), and similarly assigning all lines to a level of the PSH based on the type of connections at each end of the line. The PSH will then have ten levels.

Figure 4 shows an example of a PSH; the PSH of the drawing seen in Figure 1. Note that the outer contours of objects are judged to be more significant than the inner contours, or lines perceived as texture or pattern. (This example also shows the results of processes which fill in the gaps in lines and generate illusory contours. Both processes are discussed in later sections.)

IV. The PSH and Segmentation

The PSH can be used to create a generic segmentation algorithm, which will segment a line drawing into perceptually significant segments [13]. The output of the algorithm is a labeled version of the line drawing where each label corresponds to a separate segment. The segmentation algorithm is described in [15].

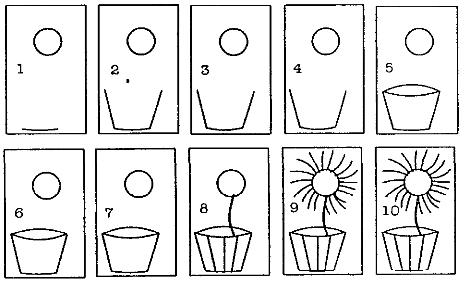

Figure 4

Figure 5 shows an example of the results of the segmentation algorithm. The two figures differ only by one small line, but we see the one as one object and the other as two. The algorithm results agree with this perception as the first figure is represented as a single segment, yet the second is represented as two segments (as indicated by the different line styles). Note that this segmentation is different from that done by Hoffman and Richards [16]. They segment a single closed contour into subparts, but the algorithm discussed here can segment an entire image into parts.

V. Rho-space Representation of Images

An important issue remains to be discussed - - is it possible to detect and represent the end connection features in images? Assume that a line drawing can be turned into a binary image. It may have lines which vary in width, which means some method of extracting a representation of lines must be used. However, standard thinning techniques cannot be used as they alter the connectivity of lines. Similarly there are many types of edge detectors that cannot be used, as they alter connectivity of lines or regions. For example, the type B end connection would never occur with circularly symmetric gaussian based edge operators. Finally, the techniques for representation must be able to provide an interpretation for intersecting lines in which the constituent lines are not necessarily connected.

A technique which meets these criteria was developed - it is the rho-space representation of oriented edges as described in Walters [13,14]. In rho-space the x and y dimensions correspond to the spatial dimensions of the image, and the rho dimension is the local orientation dimension. Rho-space is assumed to be a discretely sampled space, thus there are only a limited number of orientations explicitly represented. The x and y dimensions are also discretely sampled.

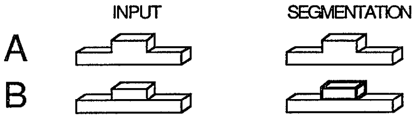

Figure 5

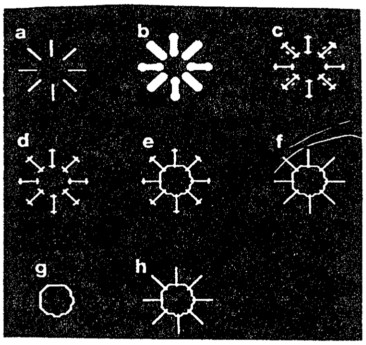

Figure 6

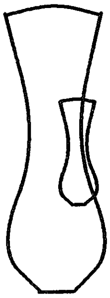

Figure 7

The algorithms developed for the rho-space representation assume there is a single processor associated with each point or pixel in space, and that each processor is locally connected only to those processors in it's local 3-D neighborhood. Thus rho-space might be implemented using a 3-D mesh connected computer.

The input to rho-space is the nonthresholded output of oriented edge operators. The basic idea is that local computations based on the value of points within a local neighborhood can be used to process an image and detect and represent the end connection features. Thus, although the response of a single edge operator cannot be unambiguously interpreted through the local interactions between neighboring responses, an unambiguous interpretation is possible.

The local parallel computations performed on the rho-space representation can be loosely described as thinning the line representations, filling short gaps in lines, and removing noise. An example of these excitatory and inhibitory processes is seen in Figure 6, where part a contains the original image. Part b shows the nonzero responses of the edge operators. Part c shows the results after lateral inhibition, which removes many spurious responses of the edge operators. Part d is after short-range linear inhibition which removes more nonzero line points. Part e shows the small gaps being filled in by short-range linear excitation, while part f shows the short, unconnected lines removed after mid-range linear inhibition, yielding the "clean" image. (More details of these excitatory and inhibitory interactions can be found in [14]).

Figure 6 also shows how these parallel algorithms have the useful property of generating illusory contours. If Figure 6a is viewed from the appropriate distance, the center may appear darker than the background. This can be explained by an illusory contour being formed around the central region. Note that in Figure 6f an illusory contour is generated as a result of the rho-space computations. Notice the short lines remain in Part c after lateral inhibition. Such lines were considered to be an inherent problem of oriented edge operators by Marr and Hildreth [17], but in fact these orthogonal end lines may play a key role in producing illusory contours. It is also interesting to note that the PSH for this image indicates that the illusory contour is more perceptually significant than the straight lines in the pattern.

VI. Application of PSH and Rho-space to Color Separation

The PSH and rho-space algorithms can be used to automate the color separation process by removing the necessity for the manual touch-up stage previously required to solve the four problems listed in Section 1.1. As some of the initial computations such as convolution are computationally intensive, they can be performed non-interactively in a preprocessing stage. After preprocessing, the resultant image can be sent to graphics workstations for interactive processing.

A. Preprocessing

The line drawings to be color separated are digitized. In order to apply the vision algorithms, the images must then be convolved with a set of oriented line detectors, to transform the image into the rho-space representation. The next stage of processing involves the local parallel operations performed on the rho-space representation.

The final stage of preprocessing is the detection and representation of instances of the end-connection features in the image. Using these features, the PSH algorithm attaches a label to each image line which indicates its perceptual significance level. These labels will be used in the interactive processing stage. The labeled image is now ready for the interactive stage of processing.

B. Interactive Color Separation

The preprocessed line-drawing images can be viewed by the user on the monitor at the graphics workstation. As a supplement to the currently available graphics techniques, the user has three interactive techniques available which are based on these computer vision algorithms:

1) PSH,

2) Segmentation, and
3) Line Extension.

The PSH can be used to solve the texture and the intersecting lines problems and to solve some contour intersection problems.

The user can select the PSH, and then by moving a slider view it at any level of significance. To color in the bottom of the container in Figure 1, the user could select the PSH shown in Figure 4. The user can then select a particular sub-image. For example, using a slider value of 5, 6, or 7, the user could select the image shown in Figure 4 and a single fill operation can be performed.

Similarly, to fill the top of the container, a value of 5, 6 or 7 could be chosen to allow the fill of the perceptual region with one command.

But not all contour intersection problems are solved by this technique. For example, with the two vase picture seen in Figure 7, all lines would appear at the 1 level. In that case, the user could select the segmentation image and then view the segments individually, which would allow filling in one step.

The rho-space algorithms can fill gaps in lines. In addition, the system can interactively extend lines or curves to fill gaps. This is done by extending each line which is not connected at its end, in a direction determined by it's local curvature. The rho-space representation makes this easily implementable. Any new end connections are detected and the associated lines temporarily relabeled with their new end-connection enhancement values. The amount that lines are extended is controlled by the user using a graphical potentiometer. Once an extended line meets another line, the line is not further extended.

Finally illusory contours can be formed by rho-space processing. These contours then become just as real as any other contour, and thus can be used for filling.

C. Success of the Vision Algorithms

It is estimated that the use of the computer vision algorithms in the interactive fake color separation process can accomplish 80% of the tasks presently done during the manual retouching stage. This means that it is now more economical to dispense with the labor-intensive retouching stage, and to handle the gap-filling, and removal of extraneous lines interactively using the vision/graphics system.

The success of these vision algorithms appears to arise from their basis in features used by the human visual system, which makes them ideally suited for dealing with human-generated line-art. Another reason for their success is that these vision algorithms are general-purpose: they do not require any model-based processing, or domain-specific knowledge, but depend instead on generic knowledge about line-art. A final reason for the success of these vision algorithms, is that they generally provide more than one technique for solving a given color separation problem, and this redundancy improves the chances of finding a solution.

References

1. Alvy Ray Smith, "Tint Fill," *SIGGRAPH Conference Proceedings*, pp. 276-283 (August 79).
2. S. Tanimoto and T. Pavlidis, "A hierarchical data structure for picture processing," *CGIP* **4,2** pp. 104-119 (June 1975).
3. D. Marr and H.K. Nishihara, "Representation and recognition of the spatial organization of three-dimensional shapes," *Proc. R. Soc. Lond.* **B-200** pp. 269-294 (1978).
4. D.K.W. Walters and N. Weisstein, "Perceived brightness is influenced by structure of line drawings," *Investigative Ophthalmology and Visual Science* **22** p. 124 (1982).
5. D.K.W.Walters, "Object interpretation using boundary based perceptually valid features," *Proc. of SPIE Applications of AI III* **635** pp. 196-202 (1986).
6. K.Paler, J.Foglein, J.Illingworth, and J.Kittler, "Local-ordered grey levels as an aid to corner detection," *Pattern Recognition* **17**(5) pp. 535-543 (1984).
7. D.A. Huffman, "Impossible objects as nonsense sentences," pp. 295-323 in *Machine Intelligence 6*, ed. D. Mitchie,Edinburgh University Press, Edinburgh (1971).
8. M.B. Clowes, "On seeing things," *Artificial Intelligence*, pp. 79-116 (2).
9. D.I. Waltz, "Understanding line drawings of scenes with shadows," in *The Psychology of Computer Vision*, ed. P.H. Winston,MaGraw-Hill, New York (1975).
10. I.Chakravarty, "A generalized line and junction labeling scheme with applications to scene analysis," *IEEE Trans. Pattern Anal. Machine Intell.* **1** pp. 202-205 (1979).
11. R.I.D.Cowie, "The viewer's place in theories of vision," IJCAI Proceedings (1983).
12. D. Lowe and T. Binford, "Segmentation and aggregation: figure ground phenomenon," Workshop on Human and Machine Vision, Montreal (1984).
13. D.K.W.Walters, "Selection of image primitives for general-purpose visual processing," *Computer Vision Graphics & Image Processing* **37**(3) pp. 261 - 298 (1987).
14. D.K.W. Walters, "Parallel Computations in Rho-Space," Proceedings of the First International Conference on Neural Networks, SanDiego, CA (June, 1987).
15. D.K.W.Walters, "Selection and use of image features for segmentation of boundary images," Proceedings IEEE CVPR (1986).
16. D.D. Hoffman and Whitman Richards, "Parts of recognition," MIT AI Memo 737 (1983).
17. D. Marr and E. Hildreth, "Theory of edge detection," *Proceedings of the Royal Society of London B* **207** pp. 187 - 217 (1980).