Stylization and Abstraction of Photographs

Doug DeCarlo Anthony Santella

Department of Computer Science & Center for Cognitive Science
Rutgers University

Abstract

Good information design depends on clarifying the meaningful structure in an image. We describe a computational approach to stylizing and abstracting photographs that explicitly responds to this design goal. Our system transforms images into a line-drawing style using bold edges and large regions of constant color. To do this, it represents images as a hierarchical structure of parts and boundaries computed using state-of-the-art computer vision. Our system identifies the meaningful elements of this structure using a model of human perception and a record of a user's eye movements in looking at the photo; the system renders a new image using transformations that preserve and highlight these visual elements. Our method thus represents a new alternative for non-photorealistic rendering both in its visual style, in its approach to visual form, and in its techniques for interaction.

CR Categories: I.3.3 [Computer Graphics]: Picture/Image Generation; I.4.10 [Image Processing and Computer Vision]: Image Representation—Hierarchical

Keywords: non-photorealistic rendering, visual perception, eye-tracking, image simplification

1 Introduction

The success with which people can use visual information masks the complex perceptual and cognitive processing that is required. Each time we direct our gaze and attention to an image, our visual intelligence interprets what we see by performing sophisticated inference to organize the visual field into coherent regions, to group the regions together as manifestations of meaningful objects, and to explain the objects' identities and causal histories [Marr 1982; Leyton 1992; Hoffman 1998; Regan 2000]. The fact that looking at a picture so often brings an effortless and detailed understanding of a situation testifies to the precision and subtlety of these inferences.

Our visual abilities have limits, of course. Good information design depends on strategies for reducing the perceptual and cognitive effort required to understand an image. When illustrations are rendered abstractly, designers can take particularly radical steps to clarify their structure. Tufte [1990] for example suggests making detail as light as possible to keep the main point of a presentation perceptually salient, and warns against adding any detail that doesn't contribute to the argument of a presentation. Thus expert illustration in instruction manuals portrays fine detail only on the object parts relevant to the current task. When artists purposely invert these heuristics, as in the popular *Where's Waldo?* pictures [Handford 1987]—which offer the visual system no salient cues to find their distinguished character—they make extracting visual information acutely demanding.

This paper describes a computational approach to stylizing and abstracting photographs that responds in explicit terms to the design goal of clarifying the meaningful visual structure in an image. Our approach starts from new image representations that recognize the visual parts and boundaries inherent in a photograph. These representations provide the scaffolding to preserve and even emphasize key elements of visual form. A human user interacts with the system to identify meaningful content of the image. But no artistic talent is required, nor even a mouse: the user simply *looks* at the image for a short period of time. A perceptual model translates the data gathered from an eye-tracker into predictions about which elements of the image representation carry important information. The simplification process itself can now apply an ambitious range of transformations, including collapsing away details, averaging colors across regions, and overlaying bold edges, in a way that highlights the meaningful visual elements. Results are shown above and in Section 5.

Since we aim for abstraction, not realism, our research falls squarely within the field of non-photorealistic rendering (NPR) [Gooch and Gooch 2001]. In the remainder of this section, we situate our approach within this field and clarify the contribution that our approach makes. Then, after a review of relevant research in human and machine vision in Section 2, we describe first our image analysis algorithm in Section 3 and then our perceptual model and simplification transformations in Section 4.

1.1 Motivations and Contributions

In the hands of talented artists, abstraction becomes a tool for effective visual communication. Take the work of Toulouse-Lautrec, in Figure 1 for example. No account of the poster, advertising the Parisian cabaret *Moulin Rouge*, could omit its exciting and memorable content. But the content commands the attention it does in no small part because of the directed, simplified organization of the poster: Toulouse-Lautrec has rendered the scene with meaningful abstraction. As observed by vision scientists such as Zeki [1999], such abstraction results in an image that directs your attention to its most meaningful places and allows you to understand the structure there without conscious effort. Such examples make sense of the aims of the field of non-photorealistic rendering, to produce abstract renderings that achieve more effective visual communication [Herman and Duke 2001].

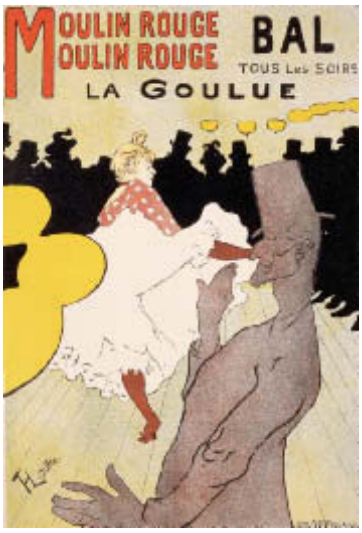


Figure 1: Henri de Toulouse-Lautrec's "Moulin Rouge—La Goulue" (Lithographic print in four colors, 1891). The organization of the contents in this poster focuses attention on the dancer La Goulue as she performs the Cancan. The use of bright, uniform colors distinguishes the figure from the background, while the placement of strokes on her dress provides rich information about its shape and material. Meanwhile, her dance partner lacks the colors (but has detail strokes), and background objects such as the spectators, are simply drawn in silhouette.

Abstraction depends on adopting a rendering style that gives the freedom to omit or remove visual information. Painterly processing, which abstracts images into collections of brush-strokes [Haeberli 1990; Litwinowicz 1997; Hertzmann 1998; Shiraishi and Yamaguchi 2000], is a notable example of such a style.

Our system transforms images into a line-drawing style using large regions of constant color; this style is very different from the painterly approaches of previous image-based work, and perhaps more closely approximates the style of printmaking of Figure 1. A similar visual style was used in the recent film *Waking Life*[1] and for producing "loose and sketchy" animation [Curtis 1999].

Once a style is in place, the key problem for interactive and automatic NPR systems is to direct these resources of style to preserve meaningful visual form, while reducing extraneous detail. Visual form describes the relationship between pictures of objects and the physical objects themselves. Painterly abstraction can cue visual form heuristically by emphasizing parts and boundaries in an image through techniques such as aligning brush strokes perpendicular to the image gradient [Haeberli 1990], terminating brush strokes at edges [Litwinowicz 1997], or drawing in a coarse-to-fine fashion [Hertzmann 1998; Shiraishi and Yamaguchi 2000]. NPR rendering methods that work from geometric models can cue visual form in more general ways, by detecting edges that arise from occluding contours or creases [Saito and Takahashi 1990; Markosian et al. 1997], and by determining appropriate directions for hatching [Hertzmann and Zorin 2000]. But models of visual form also have a fundamental role in understanding human and machine vision, and even human artistic style. For instance, Koenderink [1984b] proves that convex parts of the occluding contour of a smooth surface correspond to convexities of the surface, and that concave parts of the contour correspond to saddles; he then provides an example of Dürer's engravings that exhibit changes in hatching technique where the sign of the contour curvature changed.

Our system models visual form using state-of-the-art techniques from computer vision to identify the natural parts and boundaries in images [Comaniciu and Meer 2002; Meer and Georgescu 2001; Christoudias et al. 2002]. Our system is the first to formulate the process of abstraction completely in terms of a rich model of visual form.

Automatic techniques are more limited in their abilities to reduce extraneous detail. This is because automatic techniques cannot as of yet identify the *meaningful* elements of visual form. (Some may argue the problem will never be solved.) Selective omission is possible in specific domains. The right illumination model can eliminate distracting variations in brightness [Gooch et al. 1998]. In drawing trees, texture information can be omitted in the center of the tree, especially as it is drawn smaller [Kowalski et al. 1999; Deussen and Strothotte 2000]. The design of route maps can draw upon rules that embody how people use maps effectively [Agrawala and Stolte 2001]. For general image-based techniques the options are few; automatic painterly rendering systems simply reduce global resolution by using larger brushes [Haeberli 1990; Litwinowicz 1997; Hertzmann 1998; Shiraishi and Yamaguchi 2000]. Meaningful abstraction can only be achieved through interaction. This paper is no exception. Our approach builds upon methods for manually directing indication in pen-and-ink illustration [Winkenbach and Salesin 1994], and for controlling drawing [Durand et al. 2001] or painting [Hertzmann 2001] from an image using a hand-painted precision map. More detailed and time-consuming interaction is possible, as in the production of *Waking Life*, which relied on a combination of rotoscoping and other animation techniques.

For interaction, this paper offers a new choice of modality—eye movements [Santella and DeCarlo 2002]—and contributes new algorithms that formalize the link between fixations, perception *and visual form* to use this input effectively.

A summary of our process used to transform an image is as follows:

- Instruct a user to look at the image for a short time, obtaining a record of eye movements.
- Disassemble the image into its constituents of visual form using visual analysis (image segmentation and edge detection).
- Render the image, preserving the form predicted to be meaningful by applying a model of human visual perception to the eye-movement data.

We design the system conservatively, so that errors in visual analysis or flaws in the perceptual model do not noticeably detract from the result. Even so, manifestations of their limitations can be quite noticeable for certain images, such as those with complex textures. Nevertheless, we expect that advances in computer vision and human vision can be used directly, enabling our system to make better and richer decisions.

[1] See `http://www.wakinglifemovie.com`.

(a) (b) (c) (d)

Figure 2: (a) Original image; (b) Detected edges; (c) Color segmentation; contiguous regions of solid color in these images represent individual elements of the segmentation; (d) Color segmentation at a coarser scale (the image was first down-sampled by a factor of 16)

2 Background

2.1 Image Structure and Analysis

Our approach uses low-level visual processing to form a hierarchical description of the image to be transformed. The style of our output will be a line drawing—uniformly colored regions with black lines. We use algorithms for edge detection and image segmentation to gather the information necessary to produce such a display. There are a vast number of algorithms available for these processes; in this section, we simply describe which we are using and why. Computer vision texts (such as [Trucco and Verri 1998]) provide reviews of alternative techniques. For the remainder of this section, examples of processing will be given for the photograph in Figure 2(a), which is a 1024×768 color image.

Edge detection is the process of extracting out locations of high contrast in an image that are likely to form the boundary of objects (or their parts) in a scene. This process is performed at a particular scale (using a filter of a specific size). The Canny edge detector [Trucco and Verri 1998] is a popular choice for many applications, as it typically produces cleaner results. We use the robust variant of the Canny detector presented by Meer and Georgescu [2001], which additionally uses internal performance assessment to detect faint edges while disregarding spurious edges arising in heavily textured regions. Detected edges (using a 5×5 filter) are displayed in Figure 2(b); processing took a few seconds.

An *image segmentation* is simply a partition of an image into contiguous regions of pixels that have similar appearance, such as color or texture [Trucco and Verri 1998]. Each region has aggregate properties associated with it, such as its average color. We choose the algorithm described by Comaniciu and Meer [2002] for the robust segmentation of color images, as it produces quite clean results. Within this algorithm, colors are represented in the perceptually uniform color space $L^*u^*v^*$ [Foley et al. 1997] which produces region boundaries that are more meaningful for human observers. The parameters of this algorithm include a spatial radius h_s (similar to the radius of a filter), a color difference threshold h_r, and the size of the minimum acceptable region M. The output of this segmentation algorithm on our test image is shown in Figure 2(c) for $h_s = 7$ (in pixel units), $h_r = 6.5$ (in $L^*u^*v^*$ units), and $M = 20$ (pixels); processing took slightly over a minute.

These two algorithms can be combined together into a single system [Christoudias et al. 2002], yielding even better results; edges can be used to predict likely segmentation boundaries, and vice versa. A freely available implementation of these algorithms is available at `http://www.caip.rutgers.edu/riul`.

Scale-space theory [Koenderink 1984a; Lindeberg 1994] provides a description of images in terms of how content across different resolutions (scales) is related. This is formalized with a notion of causality: as images are blurred, smaller features come together to form larger objects so that all coarse features have a "cause" at a finer scale. This theory serves as the basis for our hierarchical representation of the image, described in Section 3. Our algorithm uses segmentation algorithms applied at a variety of scales, and finds containment relationships between their results.

2.2 Visual Perception

Our application relies on the fact that human eye movements give strong evidence about the location of meaningful content in an image. This section briefly summarizes the psychological research about the architecture of human vision that informs our work. We focus in particular on perception of static imagery.

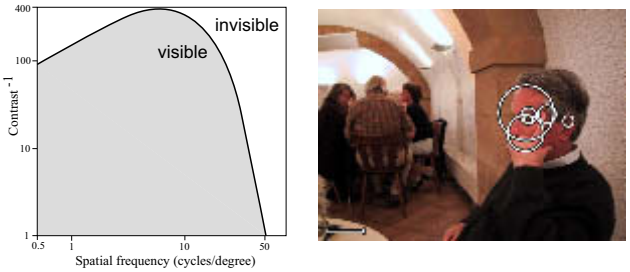
People can examine only a small visual area at one time, and so understand images by scanning them in a series of *fixations*, when the eye is stabilized at a particular point. The eye moves between these fixations in discrete, rapid movements called *saccades*, typically without conscious planning. The eye can move in other ways, such as smoothly pursuing a moving object, but the saccades and fixations are the key to understanding static images.

Fixations follow the meaningful locations in an image closely [Mackworth and Morandi 1967; Henderson and Hollingworth 1998], and their durations provide a rough estimate of the processing expended on understanding corresponding parts of the image [Just and Carpenter 1976]; fixations that land on uninteresting or unimportant objects are very short [Henderson and Hollingworth 1998]. Naturally, the information a person needs depends on their task, and fixation locations change accordingly [Yarbus 1967; Just and Carpenter 1976].

Within each fixation, the fine detail that will be visible depends on its *contrast*, its *spatial frequency* and its *eccentricity*, or angular distance from the center of the field of view [Mannos and Sakrison 1974; Koenderink et al. 1978; Kelly 1984]. Contrast is a relative measure of intensity of a stimulus, as compared to its surroundings (it is dimensionless). In psychophysical studies, the typical measure of contrast between two intensities l_1 and l_2 (with l_1 being brighter) is the Michelson contrast: $\frac{l_1 - l_2}{l_1 + l_2}$ [Regan 2000] (which is always between 0 and 1). Contrast sensitivity, which is simply the reciprocal of the contrast, is the typical measure used by psychophysicists to gauge human visual performance. Drawing on results from experiments on the perception of sine gratings (i.e. blurry stripes), Mannos and Sakrison [1974] provide a contrast sensitivity model that describes the maximum contrast sensitivity (or minimum contrast) visible at a particular frequency f (in cycles per degree):

$$A(f) = 1040(0.0192 + 0.144f)\mathrm{e}^{-(0.144\mathrm{f})^{1.1}} \qquad (1)$$

This is graphed in Figure 3(a) in log-log scale; frequency-contrast pairs in the shaded region correspond to gratings discernible to the human eye. Above this curve, the gratings simply appear as a uniform gray. This particular model has been used in graphics for



(a) (b)

Figure 3: (a) A contrast sensitivity function describes the maximum discernible contrast sensitivity as a function of frequency (of a sinusoidal grating). Values above the curve (which have low contrast) are invisible. (b) Fixations gathered using the image in Figure 2(a). Each circle has its center at the estimated location of the fixation, while its diameter indicates its duration (the scale in the lower left measures 1 second).

producing perceptually realistic images of scenes [Pattanaik et al. 1998; Reddy 2001]. Campbell and Robson [1968] adjust this model for the viewing of square-wave gratings (i.e. crisp stripes). Color contrast is a much more complicated story, however, and is not well understood [Regan 2000]. For colors that only differ in luminance, the above model applies reasonably well. When they differ further, sensitivity is typically increased. Further psychophysical studies are also required to develop better models for natural scenes as opposed to simple repeating patterns.

Contrast sensitivity decreases as a function of eccentricity [Koenderink et al. 1978]. A concise model describing this reduction [Rovamo and Virsu 1979] has been used for modeling visual acuity (which describes the highest spatial frequency that can be resolved at maximum contrast) for performance-based visualization of 3-D environments [Reddy 2001] or for deciding an appropriate brush size in painterly rendering [Santella and DeCarlo 2002]. In terms of the eccentricity angle e (in degrees), the sensitivity reduction factor $M(e)$ is 1 at the fovea center and decreases towards 0 with increasing e. The resulting contrast sensitivity function that depends on eccentricity is simply $A(f)M(e)$.

These limits on sensitivity within the visual field fit hand-in-hand with the ability of the visual system to integrate information with movements of the eyes and head. Thus, we combine information about fixation location with information about sensitivity to fine detail in making decisions about which features in an image were prominently visible to a user.

The key tool to obtain this information is an *eye-tracker* capable of sampling an observer's point of regard over time. Eye-tracker technology is steadily improving; they can now be placed in any work environment and used with just a brief calibration step. Upon viewing the image in Figure 2(a) for five seconds, our ISCAN ETL-500 eye-tracker (with an RK-464 pan/tilt camera) tracks the subject's eye movements. Corresponding fixation locations and durations are detected using a velocity threshold [Duchowski and Vertegaal 2000], and are plotted in Figure 3(b) as circles centered at the fixation location; the diameter of the circles is proportional to the duration.

Eye-trackers have seen appreciable use in human-computer interaction research. A common function is as a cursor [Sibert and Jacob 2000], either on the screen or in a virtual environment. Other roles include assessing user attention in teleconferencing systems [Vertegaal 1999], and using gaze to guide decisions for image degradation and compression [Duchowski 2000].

In our case, we use the eye-tracker indirectly in the computer interface. Our instructions are simply "look at the image"; and viewers do not have to use or attend to the eye-tracker or to their eye movements—just to the image. People are already adept in locating the desired information in images. We aim to exploit this natural ability in our interface, not to distract from it by suggesting that the user make potentially unnatural voluntary eye-movements. This paradigm still enables a computer system to draw substantial inferences about a user's attention and perception. We expect to see it used more widely.

3 Hierarchical Image Representation

Our image simplifications rest on a hierarchical representation of visual form in input images. Each image is analyzed in terms of its constituent regions by performing a *segmentation* at many different scales. We depend on regularities in *scale-space* to assemble this stack of image segmentations into a meaningful hierarchy. This section describes how we create this representation and how we extract edges using the methods described in Section 2.1. Other multiscale segmentation algorithms already exist in the vision community [Ahuja 1996]; here we draw on available code to help allow our results to be more easily reproduced [Christoudias et al. 2002].

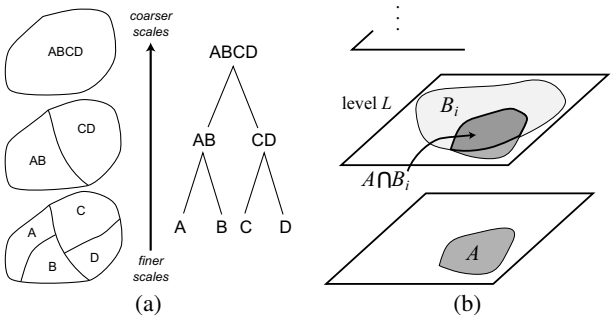
We start with an image pyramid [Burt and Adelson 1983], which is a collection of images; each one is down-sampled by a constant factor from the previous one. We use a constant factor of $\sqrt{2}$ (instead of the typical value of 2), which produces more consistency between structures across levels, and admits a simple algorithm to infer a hierarchy. A segmentation is computed for each image in the pyramid (using the parameters: $h_s = 7$, $h_r = 6.5$, $M = 20$). Figure 2(d) shows the segmentation result of an image down-sampled by a factor of 16. While the alternative of segmenting the original image at a series of increasing spatial resolutions is more faithful to scale-space, it is substantially slower.

Edges are detected in the original image using a 5×5 kernel. For this application, we have not found it necessary to detect edges at different scales. Through a process called edge tracking [Trucco and Verri 1998], detected edge pixels come together to form individual curves, which are each represented as a sequence of pixel locations. This results in a list of *edge chains*. These are the source of the curved strokes drawn in our output.

3.1 Building the Hierarchy

We now form a hierarchy starting from the regions in the segmentation of the bottom image of the pyramid (the largest image in the stack). Scale-space theory suggests regions in finer scale segmentations are typically included within regions at coarser scales (there are exceptions, however) [Lindeberg 1994]. These containments induce a *hierarchy* of image structures. Figure 4(a) shows an idealized example of such a hierarchy. Regions A, B, C and D are detected at a fine scale, where A and B combine into AB and C and D combine into CD at a coarser scale, and all combine into a single region ABCD at an even coarser scale. To represent this hierarchy, we can construct a tree (on the right of Figure 4(a)) that documents the containment relationships of regions found by segmenting at various scales. The nodes in the tree contain properties of that region, such as its area, boundary, and average color.

Noise in the images and artifacts from the segmentation prevent this from being a perfect process. Even so, we can define a hierarchy where parents are defined as the union of the areas of their children regions. In doing this, virtually all of the cases are clear-cut, allowing us to use a simple algorithm for building the hierarchy from the leaves up. For questionable situations we rely on a simple heuristic to make the choice, with the possibility of deferring a choice should it cause an invalid tree (regions must be connected). The algorithm is as follows.



(a) (b)

Figure 4: (a) A hierarchical segmentation, and its corresponding tree representation; (b) the overlap rule used to infer this hierarchy from the segmentation pyramid.

- Provided with the pyramid of segmentations, a leaf is created for each region in the bottom (finest scale) segmentation of the pyramid. Add each of these regions to the active set of regions R (which will be maintained to be those regions which currently have no parent).
- Proceeding up the pyramid, at level L:
  - For each active region $A \in R$ (which comes from a lower level), compute its best potential parent P_A. From the set of regions $\{B_i\}$ on level L which overlap with A, P_A is selected from this set as the one which maximizes:

$$\text{overlap}(A,B_i) = \frac{\text{area}(A \cap B_i)}{\left\|\text{color}(A) - \text{color}(B_i)\right\| + 1}$$

    where colors are expressed in $L^*u^*v^*$. This is depicted in Figure 4(b).
  - Assign regions to parents in order of increasing area$(A \cap P_A)$, contingent on it being connected to the children P_A already has (this prevents the formation of disconnected regions).
  - When assigned, remove A from R and add P_A (if not already present); unassigned regions remain in R.
- Of the remaining regions in R, those under 500 pixels are merged into adjacent regions (as they probably violated scale-space containment). A root region that represents the entire image parents the rest.

4 Rendering with a Perceptual Model

The rendering process works directly from the hierarchical segmentation and edge chains that were described in the last section. The output is abstracted by pruning the segmentation tree and list of edge chains: working from a set of fixations, structure is removed if the perceptual model predicts the user did not see it. This perceptual model extends our previous work [Santella and DeCarlo 2002] in two ways that rely on our new representations of visual form. First, the new model uses region structure to judge contrast sensitivity (instead of acuity). Second, it computes perceptibility of image regions rather than individual pixels. This allows the new model to be much more selective in highlighting important image parts and boundaries and in discarding extraneous detail.

With our new model, a rendering of a line-drawing using the hierarchy simply corresponds to drawing those regions on a particular frontier of the segmentation tree. (A *frontier* is a set F of nodes such that every path from the root to a leaf includes precisely one node from F.) In producing the rendering, our system smooths the boundaries of these frontier regions, draws them onto the canvas, and then overlays lines using the edge chains.

The perceptual model, which relies on eye movement data to compute eccentricities, is used to decide where to place the frontier and which lines to draw. A depth-first search defines this frontier; the access to the perceptual model is a boolean function SPLIT(n), which determines whether to draw all children of the node n based on the available fixation data (via eccentricities). The recursion proceeds by visiting a node n. If SPLIT(n) is true, then all of its children are visited. Otherwise, it is simply marked as a frontier node.

4.1 Using fixation data

Our new model interprets eye-tracking data by reference to our hierarchical description of image contents. Our raw data is a time-indexed sequence of points-of-regard measured passively by an eye-tracker as the viewer examines the image. We parse this into k fixations [Duchowski and Vertegaal 2000] $\{\mathbf{f}_i = (x_i, y_i, t_i) \mid i \in [1..k]\}$ where (x_i, y_i) are the image coordinates of the fixation point, and t_i is its duration.

In many cases, eccentricities of regions with respect to a particular fixation are solely determined using that fixation. However, estimating the *target* of each fixation enables sharp delineations of detail in the output. Each fixation $\mathbf{f}_i$ is associated with a target region n_i in the segmentation tree—n_i is represents the coherent part of the image that was viewed. We use the following method to determine n_i, as human vision research currently has little to say about this. Centered at each fixation is a circle whose size matches 5 degrees of the center of the viewer's visual field—roughly the size of their fovea [Regan 2000] (180 pixels across in our setup). We determine n_i as the smallest region that substantially overlaps this circle: there must be a set of leaf regions within n_i that are entirely inside the circle, which, taken together, comprise an area greater than half the circle and also greater than half of n_i. When no such region exists, the target cannot be identified, and n_i is set to be the leaf that contains the fixation point. The set of nodes $N = \{n_i \mid i \in [1..k]\}$ thus reports the *parts of the image* that the user looked at.

For a particular fixation $\mathbf{f}_i$ and region r, when r is either an ancestor or descendant of n_i, then its eccentricity with respect to $\mathbf{f}_i$ measures the angular distance to the closest pixel in r. Otherwise, r is assigned a constant eccentricity e_{outside} for $\mathbf{f}_i$; this provides a parameter that affects the level of content in the distant background (we use $e_{\text{outside}} = 10^\circ$). This regime induces discontinuities in estimated eccentricity at part boundaries, which means background information that is adjacent to important regions is not inappropriately emphasized, as it was in our previous approach [Santella and DeCarlo 2002].

4.2 Region Perceptibility

The pruning of the segmentation tree is based on decisions made by the perceptual model. In this model, the prominence of a region depends on its spatial frequency and contrast relative to its surroundings, as given by the contrast sensitivity threshold (see Section 2.2).

The frequency of a region is estimated as $f = \frac{1}{2D}$ [Reddy 2001], where D is the diameter of the smallest enclosing circle. In keeping with our understanding of our hierarchical structure as a representation of meaningful relations in the image, we estimate the contrast of a region by a weighted average of the Michelson contrast with its *sister* regions, where the weights are determined by the relative lengths of their common borders (this reduces to an ordinary contrast measure for regions with one sister region). In considering color contrast [Regan 2000], we use a slight variation: $\frac{\|c_1 - c_2\|}{\|c_1\| + \|c_2\|}$ (using colors in $L^*u^*v^*$). This reduces to the Michelson

contrast in monochromatic cases, and otherwise produces distances that steadily increase with perceptual differences in color.

A simple model of attention $a(t_i)$ used in our previous work [Santella and DeCarlo 2002] factors in the fixation duration t_i to scale back the sensitivity threshold. In effect, it ignores brief fixations that are not indicative of substantial visual processing, to accommodate the perceptual search required to scan a detailed image.

We are now ready to define the function SPLIT(n) for the region n. This region is split into its children if at least *half* of its children could have been perceived by any fixation. That is, a child region with frequency f, contrast c, and eccentricity e_i (for fixation $\mathbf{f}_i$) is perceptible when:

$$\frac{1}{c_{\text{scale}} \cdot c} < \max_{i \in [1..k]} \left[A\left(\max(f, f_{\text{min}})\right) \cdot M(e_i) \cdot a(t_i) \right] \qquad (2)$$

The lower bound of f_{min} (defaults to 4 cycles per degree) imposed on the frequencies used by the contrast model takes into account that low-frequency square-wave gratings are visible at lower contrasts than sine gratings [Campbell and Robson 1968] (this flattens out the left side of the curve in Figure 3(a)). To enable more substantial simplifications, we employ a contrast scaling coefficient c_{scale} to reduce contrast sensitivity (the default value is 0.1). This is a helpful parameter in fine-tuning the results, as it provides a global control for content.

4.3 Region Smoothing

Frontier regions form a partition of the image. However, the detail level of boundaries is uniformly high, since all boundaries derive from the lowest segmentation. Before rendering, the frontier region boundaries are smoothed, so that the frequencies present are consistent with the region size. Then, the regions are filled in with their average color. Figure 6(c) demonstrates the effectiveness of smoothing (showing before and after).

The network of boundaries induced by the frontier regions can be viewed as a set of curves. These curves join together at those points where three or more regions touch (or possibly two regions on the image border). Interior curves (which are not on an image border) are smoothed using a low pass filter, where the endpoints of the curve are held fixed [Finkelstein and Salesin 1994] (this preserves the network connectivity). The assigned frequency f for this curve is the maximum of the two adjoining region frequencies; this leads to use of a Gaussian kernel with $= \frac{1}{8f}$ (when filtering with this kernel, components with frequency f mostly pass through, while those at $4f$ are essentially removed). While it is possible for curves to cross each other using the filter, this is unlikely, and is unnoticeable if regions are drawn in coarse-to-fine order.

4.4 Drawing lines

With the regions drawn, the lines are placed on top. Lines are drawn using a model of visual acuity; this model ignores contrast, and instead uses the maximum perceivable frequency of $G = 50$ cycles per degree. Computing the frequency as $f = \frac{1}{2l}$ for a line of length l, the acuity model can predict it was visible if:

$$f < \max_{i \in [1..k]} \left[G \cdot M(e_i) \cdot a(t_i) \right] \qquad (3)$$

The eccentricity e_i with respect to $\mathbf{f}_i$ is determined only by the closest point on the line to the fixation point (and does not use regions in N). Lines shorter than l_{min} are not drawn. To filter out spurious lines shorter than $2.5 l_{\text{min}}$ which can appear in textured areas, we additionally require them to lie along a frontier region boundary.

Lines are smoothed in the same manner as the region boundary curves, but instead use a fixed-size filter ($= 3$). This preserves potentially important detail in lines (recall that they were extracted out using a fixed-size kernel). This also affects the placement of long lines so that they are not perfectly aligned with regions. The line thickness t depends on the length l, and is defined as the affine function which maps a range of line lengths $[l_{\text{min}}, l_{\text{max}}]$ to a range of line thicknesses $[t_{\text{min}}, t_{\text{max}}]$ (above l_{max}, thicknesses are capped at t_{max}). Lines are drawn in black, and are linearly tapered at each end over the first and last third of their length (unless the end touches an image boundary). We choose $[t_{\text{min}}, t_{\text{max}}] = [3, 10]$ and $[l_{\text{min}}, l_{\text{max}}] = [15, 500]$ as default values.

5 Results

An interaction with our system proceeds as follows. An image is selected for transformation, and is displayed on the screen in the presence of an eye-tracker. The user is instructed to "Look at the image." The image is then displayed for five seconds. In the examples that follow, all parameters are set to default values unless otherwise listed. Images and eye-movement data are available at `http://www.cs.rutgers.edu/~decarlo/abstract.html`.

We present three examples in Figures 5, 6 and 7. For each example, building the pyramid and hierarchy took about 3 minutes, and rendering took 5 to 10 seconds. The source images are displayed in (a), with the fixations that were collected by the eye tracker marked on the lower image. In each case, the line drawing that results is displayed in (b); each of these clearly exhibits meaningful abstraction. The additional renderings in Figure 7(c) illustrate the line drawing style without the use of fixation data. Instead, these drawings use a constant eccentricity in deciding whether or not to include individual regions. On the top is a drawing that maintains fine detail across the entire image, while on the bottom only coarse structures are preserved; neither clearly contains an obvious subject. This demonstrates our interactive technique: tracking eye movements enables meaningful abstraction. However, the images in Figure 7(b) and (c) are still clearly produced using the same style.

6 Discussion

In this paper we have presented a new alternative for non-photorealistic rendering, encompassing: a new visual style using bold edges and large regions of constant color; a new approach to visual form for rendering transformations on images, a hierarchical structure that relates the meaningful parts in an image across scales; and new techniques for interaction, based on eye-tracking and models of perception.

Future research can bring improvements to any of these areas. For example, the segmenter could be enhanced to use a model of shading. This would reduce the patchiness seen in smoothly shaded regions (skin, in particular). More difficult, however, is the appropriate placement of boundaries to indicate shading changes or gradations. The treatment of texture offers a stylistic challenge. Currently, simple textures are simply smoothed away (such as the stucco wall in the opening figure). Complex textures are problematic (especially when foreshortened), such as the pattern of windows in Figure 8. In this case, the segmenter lumps together all of the small windows into a single region. While our current segmenter does not model texture, other computer vision research has looked at grouping regions into textured objects. But how can a system effectively convey an inferred texture in an abstracted way?

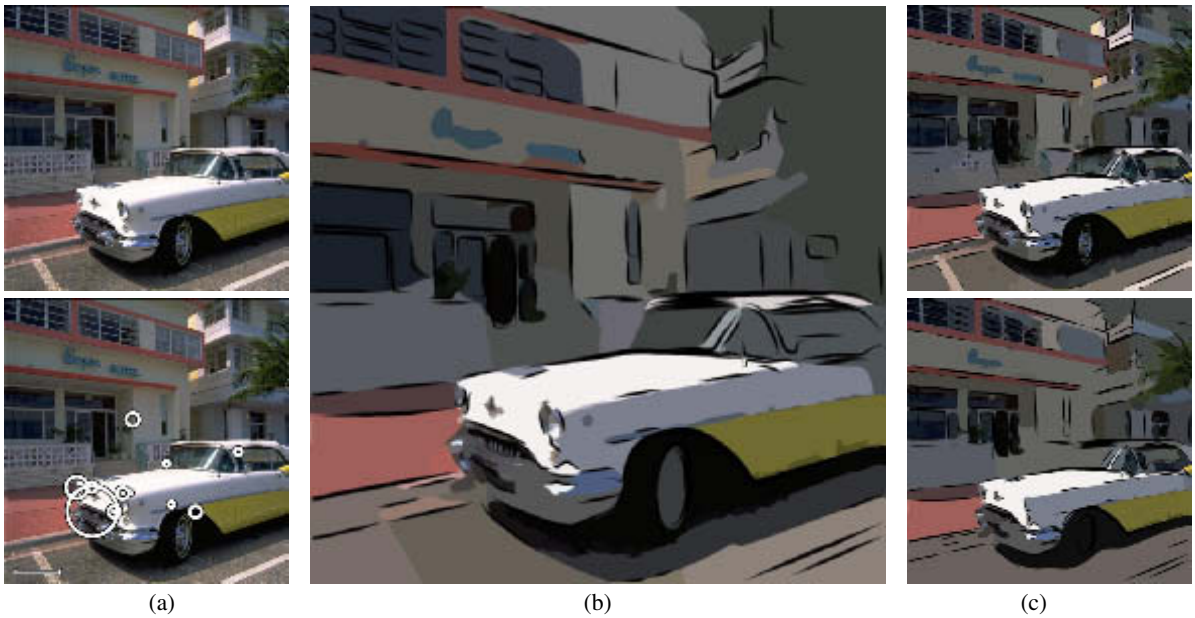
The segmentation could be enriched with additional aspects of visual form as well. Natural possibilities include the grouping of parts into coherent objects, or the status of contours as results of occlusion, shadow or markings. For animation, algorithms for the visual tracking of image features and the segmentation of moving objects will be required to achieve consistency of elements over

(a) (b)

Figure 5: (a) A source image (1024×688) and fixations gathered by the eye-tracker; (b) the resulting line drawing ($c_{\text{scale}} = 0.1$, $l_{\text{min}} = 40$).

(a) (b) (c)

Figure 6: (a) A source image (1024×688) and fixations gathered by the eye-tracker; (b) the resulting line drawing ($c_{\text{scale}} = 0.05$, $l_{\text{min}} = 40$); (c) region boundaries before and after smoothing.



(a) (b) (c)

Figure 7: Comparison with and without eye-tracking data for the 768×768 image in (a). The drawing in (b) uses fixation data, and important details (as seen by the user) are retained ($e_{\text{outside}} = 40^\circ$). The drawings in (c) instead use a constant eccentricity (3° on top, 12° on the bottom image) across the entire image so that no meaningful abstraction is performed. (All use $c_{\text{scale}} = 0.14$, $l_{\text{min}} = 15$.)

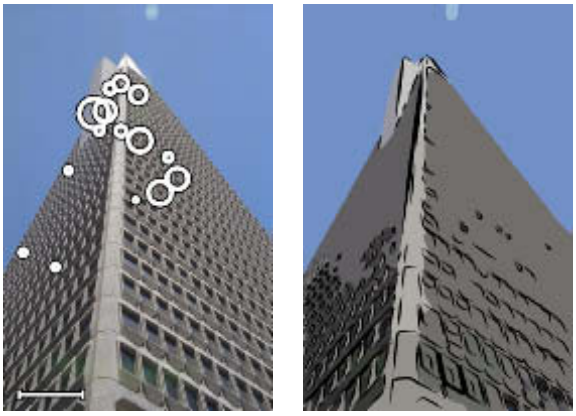


Figure 8: A photograph with a difficult texture, and its corresponding line drawing ($c_{\text{scale}} = 0.175$, $l_{\text{min}} = 15$).

time. But how can this be related to the patterns of fixations gathered across a series of images (whether they are viewed frame-by-frame or at full speed)?

Finally, more sophisticated models of visual perception can support more accurate decisions of what simplifications are possible, and suggest more discriminating transformations on regions. Indeed, by providing a controlled means for adapting imagery based on a perceptual model, our system may itself serve as a tool for formulating and testing such perceptual models.

Acknowledgments

Thanks to Eileen Kowler, Manish Singh, Peter Meer, Jan Koenderink, John Henderson, Chris Christoudias, Vašek Chvátal, Cassidy Curtis, and Matthew Stone. Photos in Figures 5-8 courtesy `http://philip.greenspun.com`. Partially supported by NSF Instrumentation 9818322.

References

AGRAWALA, M., AND STOLTE, C. 2001. Rendering effective route maps: improving usability through generalization. In *Proc. of ACM SIGGRAPH 2001*, 241-249.

AHUJA, N. 1996. A transform for multiscale image segmentation by integrated edge and region detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence 18*, 12, 1211-1235.

BURT, P., AND ADELSON, E. 1983. The Laplacian pyramid as a compact image code. *IEEE Trans. on Communications 31*, 4, 532-540.

CAMPBELL, F., AND ROBSON, J. 1968. Application of Fourier analysis to the visibility of gratings. *Journal of Physiology 197*, 551-566.

CHRISTOUDIAS, C., GEORGESCU, B., AND MEER, P. 2002. Synergism in low level vision. In *Proc. ICPR 2002*.

COMANICIU, D., AND MEER, P. 2002. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence 24*, 5.

CURTIS, C. 1999. Non-photorealistic animation. In *ACM SIGGRAPH 1999 Course Notes #17 (Section 9)*.

DEUSSEN, O., AND STROTHOTTE, T. 2000. Computer-generated pen-and-ink illustration of trees. In *Proc. of ACM SIGGRAPH 2000*, 13-18.

DUCHOWSKI, A., AND VERTEGAAL, R. 2000. Eye-based interaction in graphical systems: Theory and practice. In *ACM SIGGRAPH 2000 Course Notes #5*.

DUCHOWSKI, A. 2000. Acuity-matching resolution degradation through wavelet coefficient scaling. *IEEE Trans. on Image Processing 9*, 8 (Aug.), 1437-1440.

DURAND, F., OSTROMOUKHOV, V., MILLER, M., DURANLEAU, F., AND DORSEY, J. 2001. Decoupling strokes and high-level attributes for interactive traditional drawing. In *Proceedings of the 12th Eurographics Workshop on Rendering*, 71-82.

FINKELSTEIN, A., AND SALESIN, D. 1994. Multiresolution curves. In *Proc. of ACM SIGGRAPH 94*, 261-268.

FOLEY, J., VAN DAM, A., FEINER, S., AND HUGHES, J. 1997. *Computer Graphics: Principles and Practice, 2nd edition*. Addison Wesley.

GOOCH, B., AND GOOCH, A. 2001. *Non-Photorealistic Rendering*. A K Peters.

GOOCH, A. A., GOOCH, B., SHIRLEY, P., AND COHEN, E. 1998. A non-photorealistic lighting model for automatic technical illustration. In *Proc. of ACM SIGGRAPH 98*, 447-452.

HAEBERLI, P. 1990. Paint by numbers: Abstract image representations. In *Proc. of ACM SIGGRAPH 90*, 207-214.

HANDFORD, M. 1987. *Where's Waldo?* Little, Brown and Company.

HENDERSON, J. M., AND HOLLINGWORTH, A. 1998. Eye movements during scene viewing: An overview. In *Eye Guidance in Reading and Scene Perception*, G. Underwood, Ed. Elsevier Science Ltd., 269-293.

HERMAN, I., AND DUKE, D. 2001. Minimal graphics. *IEEE Computer Graphics and Applications 21*, 6, 18-21.

HERTZMANN, A., AND ZORIN, D. 2000. Illustrating smooth surfaces. In *Proc. of ACM SIGGRAPH 2000*, 517-526.

HERTZMANN, A. 1998. Painterly rendering with curved brush strokes of multiple sizes. In *Proc. of ACM SIGGRAPH 98*, 453-460.

HERTZMANN, A. 2001. Paint by relaxation. In *Computer Graphics International*, 47-54.

HOFFMAN, D. D. 1998. *Visual intelligence: how we create what we see*. Norton.

JUST, M. A., AND CARPENTER, P. A. 1976. Eye fixations and cognitive processes. *Cognitive Psychology 8*, 441-480.

KELLY, D. 1984. Retinal inhomogenity: I. spatiotemporal contrast sensitivity. *Journal of the Optical Society of America A 74*, 1, 107-113.

KOENDERINK, J. J., M.A. BOUMAN, A. B. D. M., AND SLAPPENDEL, S. 1978. Perimetry of contrast detection thresholds of moving spatial sine wave patterns. II. the far peripheral visual field (eccentricity 0-50). *Journal of the Optical Society of America A 68*, 6, 850-854.

KOENDERINK, J. J. 1984. The structure of images. *Biological Cybernetics 50*, 363–370.

KOENDERINK, J. J. 1984. What does the occluding contour tell us about solid shape? *Perception 13*, 321-330.

KOWALSKI, M. A., MARKOSIAN, L., NORTHRUP, J. D., BOURDEV, L., BARZEL, R., HOLDEN, L. S., AND HUGHES, J. 1999. Art-based rendering of fur, grass, and trees. In *Proc. of ACM SIGGRAPH 99*, 433-438.

LEYTON, M. 1992. *Symmetry, causality, mind*. MIT Press.

LINDEBERG, T. 1994. *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers.

LITWINOWICZ, P. 1997. Processing images and video for an impressionist effect. In *Proc. of ACM SIGGRAPH 97*, 407-414.

MACKWORTH, N., AND MORANDI, A. 1967. The gaze selects informative details within pictures. *Perception and Psychophysics 2*, 547-552.

MANNOS, J. L., AND SAKRISON, D. J. 1974. The effects of a visual fidelity criterion on the encoding of images. *IEEE Trans. on Information Theory 20*, 4, 525-536.

MARKOSIAN, L., KOWALSKI, M. A., TRYCHIN, S. J., BOURDEV, L. D., GOLDSTEIN, D., AND HUGHES, J. F. 1997. Real-time nonphotorealistic rendering. In *Proc. of ACM SIGGRAPH 97*, 415-420.

MARR, D. 1982. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. W.H. Freeman, San Francisco.

MEER, P., AND GEORGESCU, B. 2001. Edge detection with embedded confidence. *IEEE Trans. on Pattern Analysis and Machine Intelligence 23*, 12, 1351-1365.

PATTANAIK, S. N., FERWERDA, J. A., FAIRCHILD, M. D., AND GREENBERG, D. P. 1998. A multiscale model of adaptation and spatial vision for realistic image display. In *Proc. of ACM SIGGRAPH 98*, 287-298.

REDDY, M. 2001. Perceptually optimized 3D graphics. *IEEE Computer Graphics and Applications 21*, 5 (September/October), 68-75.

REGAN, D. 2000. *Human Perception of Objects: Early Visual Processing of Spatial Form Defined by Luminance, Color, Texture, Motion and Binocular Disparity*. Sinauer.

ROVAMO, J., AND VIRSU, V. 1979. An estimation and application of the human cortical magnification factor. *Experimental Brain Research 37*, 495-510.

SAITO, T., AND TAKAHASHI, T. 1990. Comprehensible rendering of 3-D shapes. In *Proc. of ACM SIGGRAPH 90*, 197-206.

SANTELLA, A., AND DECARLO, D. 2002. Abstracted painterly renderings using eye-tracking data. In *Proc. of the Second International Symp. on Non-photorealistic Animation and Rendering (NPAR)*.

SHIRAISHI, M., AND YAMAGUCHI, Y. 2000. An algorithm for automatic painterly rendering based on local source image approximation. In *Proc. of the First International Symp. on Non-photorealistic Animation and Rendering (NPAR)*, 53-58.

SIBERT, L. E., AND JACOB, R. J. K. 2000. Evaluation of eye gaze interaction. In *Proc. CHI 2000*, 281-288.

TRUCCO, E., AND VERRI, A. 1998. *Introductory Techniques for 3-D Computer Vision*. Prentice-Hall.

TUFTE, E. R. 1990. *Envisioning Information*. Graphics Press.

VERTEGAAL, R. 1999. The gaze groupware system: Mediating joint attention in mutiparty communication and collaboration. In *Proc. CHI '99*, 294-301.

WINKENBACH, G., AND SALESIN, D. H. 1994. Computer-generated pen-and-ink illustration. In *Proc. of ACM SIGGRAPH 94*, 91-100.

YARBUS, A. L. 1967. *Eye Movements and Vision*. Plenum Press.

ZEKI, S. 1999. *Inner Vision: An Exploration of Art and the Brain*. Oxford Univ. Press.