Artistic Minimal Rendering with Lines and Blocks

Paul L. Rosin [a] Yu-Kun Lai [a]

[a] *School of Computer Science and Informatics, Cardiff University, UK*

Abstract

Many non-photorealistic rendering techniques exist to produce artistic effects from given images. Inspired by various artists, interesting effects can be produced by using a minimal rendering, where the minimum refers to the number of tones as well as the number and complexity of the primitives used for rendering. Our method is based on various computer vision techniques, and uses a combination of refined lines and blocks (potentially simplified), as well as a small number of tones, to produce abstracted artistic rendering with sufficient elements from the original image. We also considered a variety of methods to produce different artistic styles, such as colour and two-tone drawings, and use semantic information to improve renderings for faces. By changing some intuitive parameters a wide range of visually pleasing results can be produced. Our method is fully automatic. We demonstrate the effectiveness of our method with extensive experiments and a user study.

Key words: non-photorealistic rendering, line drawing, tonal blocks, image abstraction

1 Introduction

Image based non-photorealistic rendering (NPR) has proliferated over the last few years, and applications such as Photoshop contain numerous effects for modifying images. NPR not only provides a more artistic style but also gives a more abstracted presentation of image (or video) contents such that the burden of perceptual understanding in communication is reduced. Many of these effects operate on a region basis (e.g. mosaic [1], stained glass [2]), build up the output image by applying a vast number of strokes (e.g. [3–6]), or apply low level filters to produce a variety of effects (e.g. [7,8]). However, while such approaches are effective, they are limited in terms of their ability to perform substantial simplification whilst retaining the essence of the image. Generating a *sparse* (and possibly abstracted or caricatured) rendering of an image such as the line drawings by Matisse or Picasso is a more challenging task.

Email addresses: Paul.Rosin@cs.cardiff.ac.uk (Paul L. Rosin),

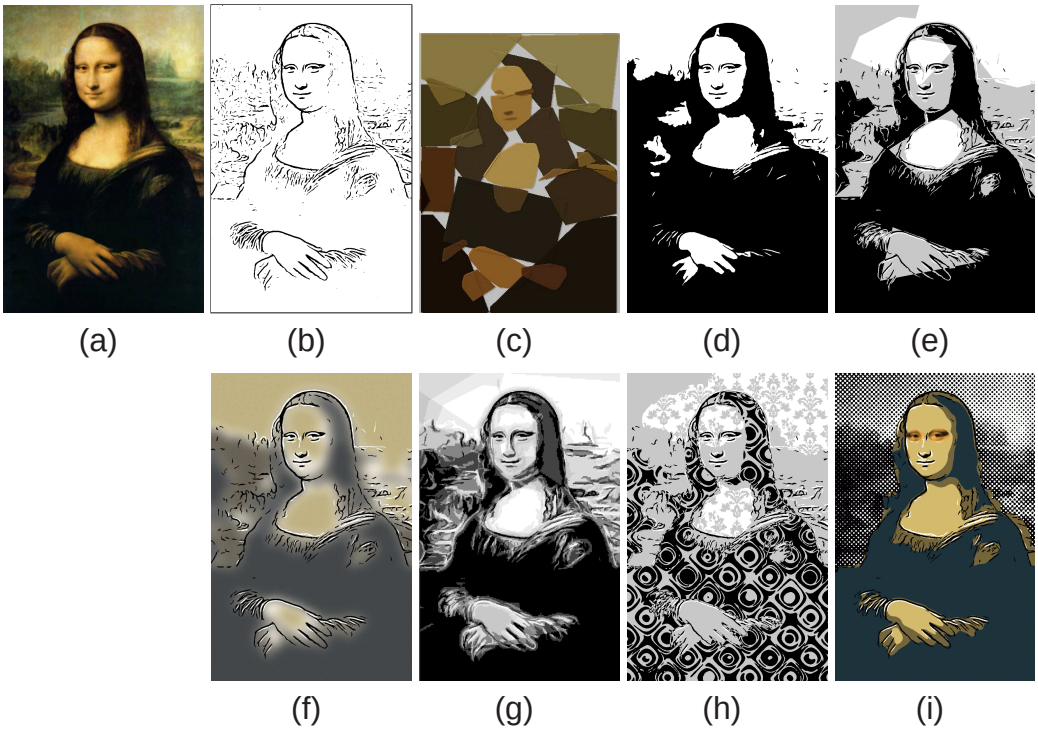

Fig. 1. Mona Lisa rendered in different styles. a) original, b) line drawing, c) image abstraction [9], d)-i): six styles of the proposed approach in this paper (2 tone, 3 tone, lines & diffuse colour, pyramid with abstracted tonal blocks, textured tonal blocks, dithered background with lines, flat colour and eye cut-outs).

In this paper we continue with the traditional bottom-up approach, and in particular take Kang *et al.* [10] as our starting point. They extract lines by applying a difference of Gaussians (DoG) filter followed by thresholding. Their contribution is to reduce fragmentation and extract highly coherent lines by adaptively determining the shape and orientation of the filter kernel from the local image characteristics. The DoG is applied in the direction perpendicular to the largest intensity contrast, as specified by the local edge flow. In turn, this is determined by performing non-linear vector smoothing of the edge tangents, using a bilateral filtering type approach, so that strong edge directions are preserved while weak edge directions are modified to follow dominant edges. At each pixel in the image the curvilinear centre line of the kernel is created by following the local edge flow a fixed distance upstream and downstream. The second dimension of the kernel is formed by expanding the centre line in the direction perpendicular to the edge flow. The line drawings produced by Kang *et al.*'s method can be attractive, but as with the previously cited methods, they cannot be directly modified to perform substantial abstraction, as shown in figure 1b.

In contrast, the work by Song *et al.* [9] is specifically designed to produce highly abstract images, reminiscent of the paper cutouts by Matisse. An example is given

Yukun.Lai@cs.cardiff.ac.uk (Yu-Kun Lai).

in figure 1c. The image is segmented at multiple scales using the normalised cut algorithm and a variety of simple shapes (e.g. circles, triangles, squares, superellipses and so on) are fitted to each region. The system automatically chooses the shape that best represents the region; the choice is made via a supervised classifier so the "best shape" depends on the subjectivity of a user. A consequence of this approach (and many others that aim for high levels of abstraction [11]) is that it is unable to capture the nuances necessary to produce, for instance, an identifiable portrait.

The aim of this paper is to use Kang *et al.*'s method to provide the detailed content from an image in the form of coherent lines, and to combine it with dark and light blocks that capture the overall tonal balance of the image, and which, inspired by Song *et al.*, are extracted from the image and then simplified and rendered in various ways. By combining abstraction along with a limited amount of readily recognisable detail, we can produce effects reminiscent of Warhol's 1980's portraits which mixed and overlaid line drawings, photographs and coloured geometric blocks such as rectangles. In addition, we consider applying some decorative effects to the blocks, in a manner inspired by Matisse who sometimes included a field of patterns overlaid on a flat background. Combining dark and light lines over a gray background has been shown to effectively capture 3D shape [12,13], in which ridge detection methods are used to extract the lines, and combined with toon shading. These works however are based on rendering input 3D objects while our work takes images as input which are more widely available.

We also consider improving the rendering by incorporating semantic information to ensure that faces (which are considered highly salient for human perception) are treated correctly. Not only does this enable more flexible treatment of the rendering, but allows us to direct the low level image processing to produce more visually pleasing results.

This paper is a substantial extension of our previous conference version [14] with significant extended algorithms mainly in section 3 (automatic model selection, double response suppression, colour and two tone rendering, face aware rendering) and additional experiments including a user study in section 4.

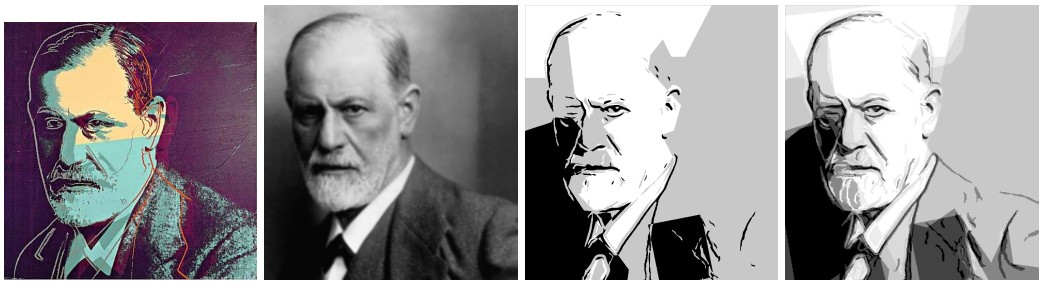

Fig. 2. Portrait of Sigmund Freud. a) version by Andy Warhol, b) photograph used as input for our rendering, c) 3 tone rendering with abstracted tonal blocks, d) 10 tone rendering with abstracted tonal blocks.

As an example, several different rendering styles of the Mona Lisa have been given

in figure 1(d-i), demonstrating the variety of effects that we can produce. A further example is given in figure 2; the silkscreen print portrait of Sigmund Freud made by Andy Warhol in 1980 is shown in figure 2a. Starting from a similar photograph (figure 2b) our algorithm also produces a rendering containing lines and blocks (figure 2c). [1]

Although the concept has been recognised for a decade [15], to date, there have been few image based minimal rendering techniques. The two most notable are the recent methods by Mould and Grant [16] and Xu and Kaplan [17], which are both based on segmentation to create an initial binarised image that is subsequently refined. A more specialised technique was presented by Meng *et al.* [18] who produce a paper-cut style rendering for human faces. This consists of a two tone output, with the foreground consisting of a single connected region. Our method also starts with regions, but then combines them with a sparse set of coherent lines. This, along with an additional mid-gray tone ensures that sufficient detail is included such that the regions can become highly abstracted whilst the rendering still recognisably resembles the original image. Most recently, Winnemoller *et al.* [19] describe an extended version of Kang *et al.*'s method; namely XDoG. The DoG is modified so that the strength of the inhibitory effect of the larger Gaussian is allowed to vary according to some parameter, and the output is further modified by an adjustable thresholding function that allows soft thresholding. In total there are seven basic parameters that control aspects such as the strength of the edge sharpening effect, contrast etc. This enables a variety of styles to be generated, such as pastel, woodcut, and, most similar to our results, a soft binary rendering which can include negative lines. However, by using explicit region representation our proposed method has more flexibility regarding abstraction, and moreover uses a fixed set of parameter values.

There is some related work that uses 3D models to generate minimal renderings. Buchholz *et al.* [20] use segmentation of 3D models in image space to generate abstracted binary renderings, while Bronson *et al.* [21] generate binary stencils (self connected sheets). Both methods focus on regions; compared with the image based methods they can better capture the object's shape given the available 3D information.

The contribution of the paper is to introduce a novel insight of a class of artistic rendering as a perceptually minimal rendering model. Rather than a unique and precisely quantifiable mathematical solution, our concept of minimality leads to a variety of renderings which aim to balance multiple factors, including recognisability, aesthetics, and number and complexity of elements. In practise it is not possible to exactly quantify these factors, and so we seek to achieve practical solutions. A variety of techniques from computer vision have been incorporated to approach this

[1] Since the contrast is so low on the left hand side of Freud's face and body, we are unable to recover some details (unlike Warhol who manually overlaid lines to reconstruct details missing from the photograph).

goal. More specifically, a careful combination of refined lines and blocks, together with a small number of tones have been combined to produce visually pleasing and recognisable artistic renderings with recognisable information using a set of "least complicated" primitives. During this process, some aesthetic criteria are also naturally taken into account, e.g. tonal balance is retained since significant dark or light regions are reproduced using blocks in the output image. This is related to the work by Rivotti *et al.* [22] where combinations of different primitives are used for aesthetic non-photorealistic rendering from 3D models.

2 Basic Method

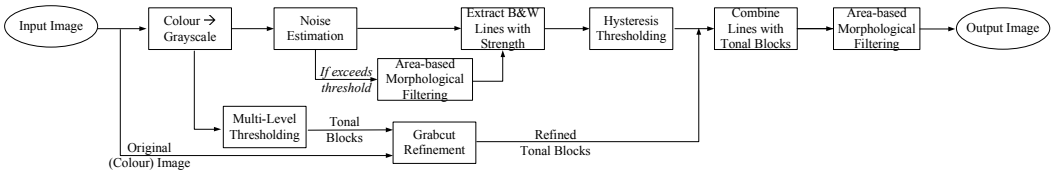

Fig. 3. Basic algorithm pipeline.

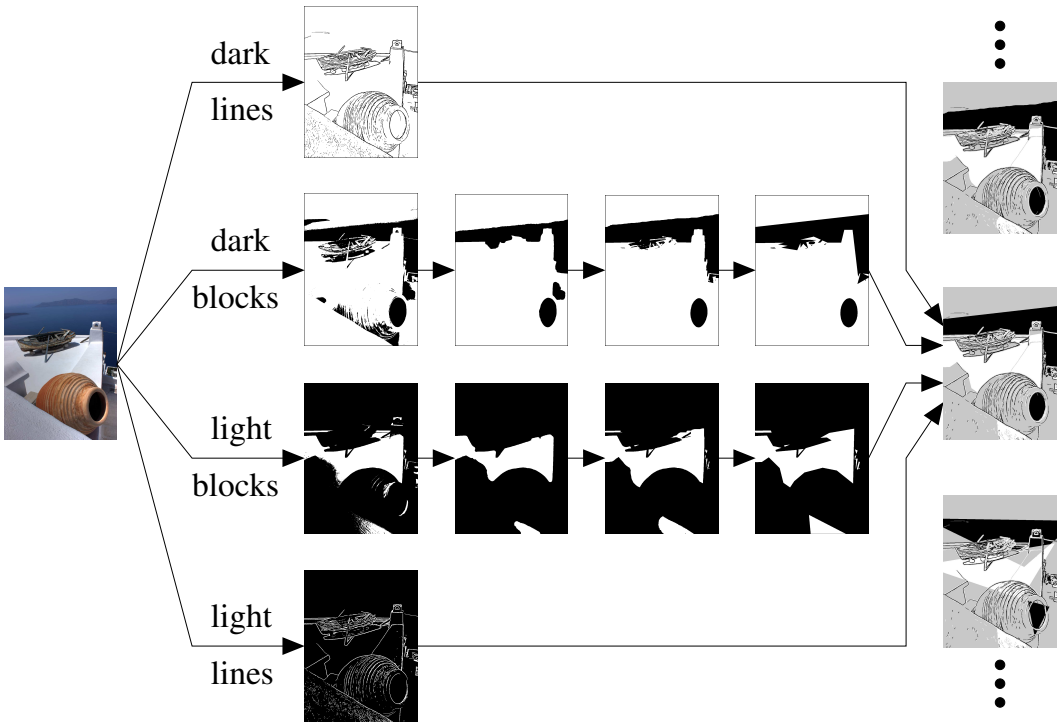

Fig. 4. Resulting images at different stages of the algorithm pipeline. The middle section shows how the initial tonal blocks (first column) are cleaned up using mathematical morphological filtering (second column), grabcut (third column), and simplification (fourth column). The tonal blocks and lines can be combined to produce a variety of styles (rightmost section).

The motivation of our algorithm is to model the open problem of artistic rendering

as using a set of "least complicated" primitives to approximate the input image, while maintaining the rendered image as recognisable as possible. We use dark and light blocks (potentially simplified) to lay out the overall tone balance. A small number (typically between 3 and 10) of tones is used instead of two as our observation shows that much more information can be preserved, using only the small perceptual overhead of one or a few extra tones. Refined and simplified coherent lines are further overlaid to enhance the visual richness while keeping the perceptual overhead well controlled.

The pipeline of the basic algorithm is given in figure 3, with figure 4 showing the resulting images at different stages. If the input is a colour image, we first convert it to grayscale. To avoid fragmented lines being produced in the output rendering for noisy images, we estimate the noise level using [23] and apply denoising if this exceeds some threshold. Hysteresis thresholding [24] is applied to the edgels to reduce fragmentation and enhance coherence. Area-based morphological opening and closing operations [25] are used to produce smoother grayscale images. This has been used for pre-processing noisy images and post-processing combined line and tonal block images. The essential components are discussed in detail later this section, including three tone drawing and extraction of refined dark and light blocks. We will discuss further improvements/extensions in the next section.

2.1 Three tone drawings

Kang *et al.*'s method was developed to draw both black and white lines on a gray background. Adding white lines allows highlights to be included, and can substantially enhance the impression of three dimensional structure. The effectiveness of this approach is well known to traditional artists, often appearing, for example, as chalk and charcoal drawings. An alternative way to indicate three dimensional shape would be to use cross hatching, but our philosophy of minimal rendering favours the approach we have taken of maintaining a small number of strokes with only the small perceptual overhead of one extra tone.

Processing is straightforward: lines are extracted from both the image and its inverted version, and then combined such that at each pixel the output tone is black (white) if there only exists a black (white) line. Otherwise (if no lines or both black and white lines are present at that pixel) the output tone is gray. This can be computed using the following $\frac{lines(I)+\overline{lines(\overline{I})}}{2}$ where I is the input image, $lines(\cdot)$ is the tone output from the line detector, and bars denote inversion. (In fact, for better perceptual effect we prefer to remap the midgray intensity to 20% density.)

Figure 5 demonstrates how an additional tone enables extra details to be picked up (e.g. along the nose and lips) that are missed in the black and white drawing. In comparison with an optimally three tone thresholded version [26] our three tone drawing provides a clearer, more readily identifiable depiction.

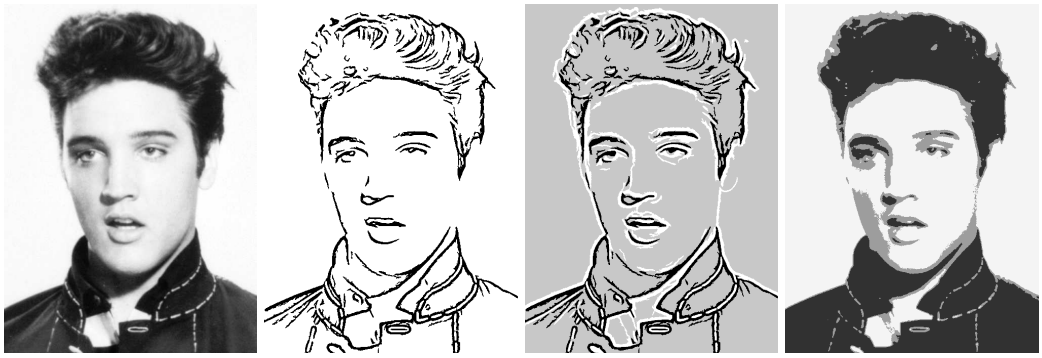

Fig. 5. Three tone renderings: a) original image, b) two tone drawing, c) three tone drawing, d) image posterised into three levels

2.2 Dark and light blocks

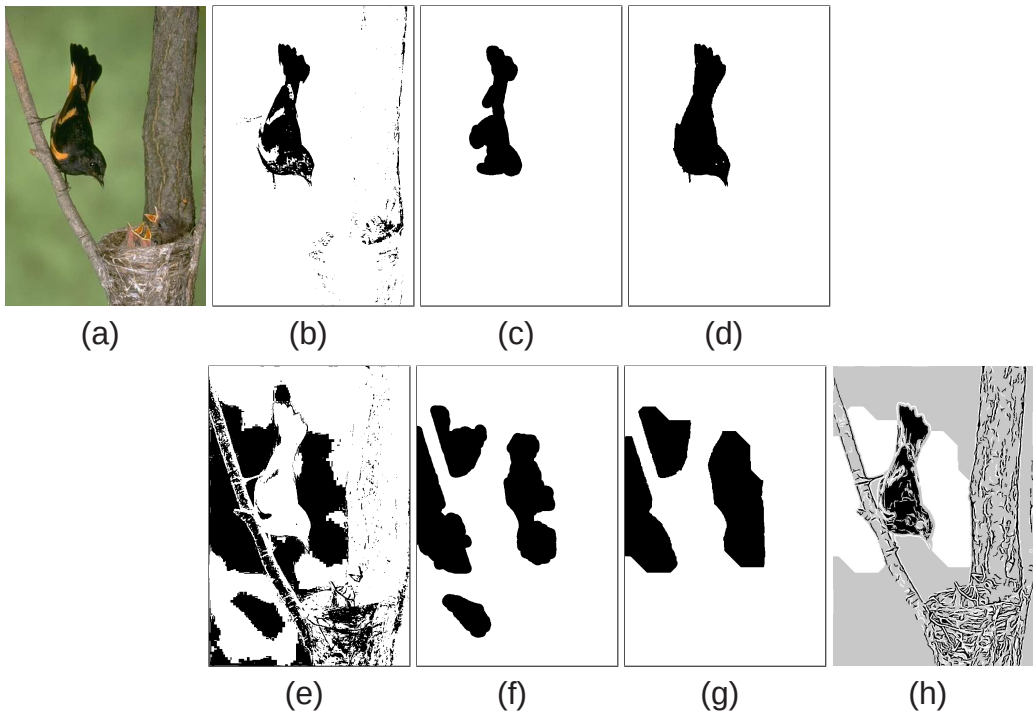

Fig. 6. Extracting masks a) original image, b) dark mask from thresholding, c) dark mask after cleaning by mathematical morphology, d) dark mask after grabcut, e) light mask from thresholding, f) light mask after cleaning by mathematical morphology, g) light mask after grabcut, h) final rendering.

For some images their nature is better captured by the rendering if their overall tonal balance is approximately retained. For instance, perceptual studies have demonstrated that when subjects are shown line drawings of faces, the inclusion of dark and light regions greatly improved identification [27]. Therefore we wish to extract and render regions from the image that represent significant tonal blocks. We consider these blocks to be dark and light regions which we would like to

locate automatically. Unfortunately, automatic and reliable image segmentation is in general an unsolved problem – thus whatever scheme we use we can expect some errors. With traditional NPR techniques that employ a huge number of individual strokes, individual erroneous strokes are hidden amongst the mass of their neighbours. Given our sparse rendering, errors, especially with regard to blocks, are much more exposed. Nevertheless, we will later describe some strategies for reducing the effects of such artifacts.

We note that the combination of lines and tonal blocks was originally applied to the rendering of faces [28] for data compression. The thresholded image intensities were combined with the extracted lines using a logical AND such that the dark tonal blocks were retained while light tonal blocks were ignored. More recently a similar approach was taken by Gooch *et al.* [29] who stated that the intensity threshold value was "chosen manually according to taste". In contrast, we desire an automatic method which is reasonably reliable and accurate, and can also be applied to a wider range of image types beyond just faces with uncluttered backgrounds.

2.2.1 Tonal block extraction

Since we want dark and light regions, image thresholding seems a reasonable starting point. Hundreds of algorithms exist [30] and we choose the standard algorithm by Otsu [31] which determines the threshold that minimises the within class variance of the two groups of pixels separated by the thresholding operator. To extract significant tonal blocks, an efficient implementation [26] of a multi-threshold variant is used; i.e. n thresholds are selected to minimise the within class variance across the $n+1$ groups of pixels. If two thresholds are selected this would generate three classes of pixels which could be considered as dark, medium intensity, and light. However, given the inevitable thresholding errors that will occur for some images, we take a more conservative approach and select three thresholds so that the dark and light classes are separated by a double class of intermediate intensity pixels. The rationale is that under-representing the dark and light blocks is preferable to over representing them.

The thresholding provides a global histogram based criterion for classifying pixels into tonal classes. This typically results in noisy and fragmented dark and light blocks which need to be cleaned up. Two steps of post processing are employed which take local spatial information into account. First, a standard mathematical morphological opening and closing is applied which removes isolated small, or thin regions. A sufficiently large structuring element is used to ensure that the majority of spurious clutter is removed, and consequently the remaining regions are substantially distorted. The purpose of the second step is to rectify this, and moreover improve the segmentation of the tonal blocks. Colour information is often used in image segmentation [32] instead of just intensity. In our case we use a segmentation technique for refinement of the tonal blocks. Grabcut [33], initialised by the output of the mathematical morphological operations, provides an effective solution. As-

sume the tonal region (either dark or light) is denoted as R. We use morphological dilation and erosion operators with a fixed width (typically chosen as 50 pixels) to obtain expanded or shrunk regions, denoted as R_+ and R_- respectively, to initialise the grabcut model with $\overline{R_+}$ as background, R_- as foreground and $R_+ - R_-$ as the unknown region to be optimised. Note that certain pixels having similar intensity may be well separated in the 3-dimensional colour space. Having the boundary optimised to snap to significant boundaries in the colour image is thus much more reliable. Colour information is also used in the generation of foreground and background Gaussian mixture models used internally in the grabcut algorithm. Thus colour similarity helps to produce refined masks taking both tonal levels and the colour similarity into account. The grabcut approach also allows the topology of the masks to be changed, i.e. some holes in the masks may be automatically added or removed. The process of extracting the masks is illustrated in figure 6.

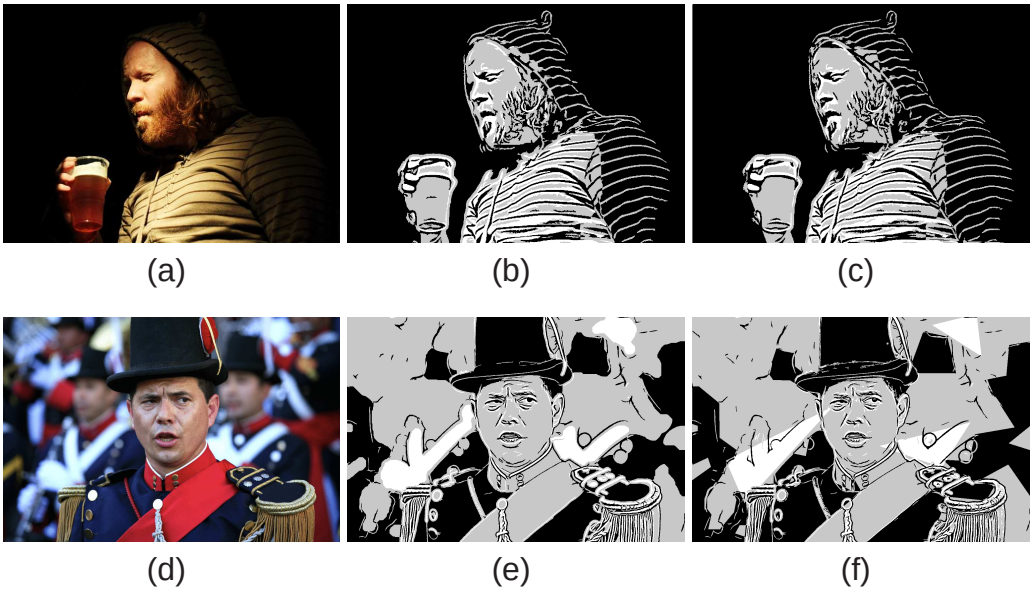

Fig. 7. Dark and light tonal blocks combined with dark and light lines (b and e), plus the blocks simplified by models selected using BIC (c and f).

2.2.2 Model fitting

Having extracted accurately delineated blocks, they can now be rendered as they are, or simplified and modified for artistic effect. As with the stylisation in [9] simple geometric primitives can be fitted to the blocks using the invariant fitting method of Voss and Süße [34] – we have used triangles, parallelograms and ellipses. Alternatively, rather than using fixed shapes, the blocks can be simplified using polygonal approximation [35].

2.2.3 *Automatic model selection*

The user can choose their preferred block stylisation method; alternatively, this task can be performed automatically for each tonal block. Previously, Song *et al.* [9] used a decision tree, but this required a cumbersome training process. In this work this is avoided by using the Bayes information criterion (BIC), which is well suited for this task since it is designed for selecting an appropriate model from a set with different numbers of parameters, as is the case for our different types of geometric models. The model selection process needs to balance the goal of reducing the fitting errors by also penalising the complexity (i.e. the numbers of parameters) of the model to avoid overfitting. This is done by minimising

$$\text{BIC} = -2\ln(L) + k\ln n$$

where L is the likelihood function for the estimated model, n is the number of data points, k is the number of free parameters to be estimated. If we assume that the residuals are normally distributed then

$$\text{BIC} = n\ln(\sigma^2) + k\ln n,$$

where σ^2 is the variance of the error (i.e. the mean summed squared residuals).

Figure 7 demonstrates the above process of BIC guided model selection. The extracted tonal blocks are shown in figure 7b&e. It can be seen that after simplification the blocks are represented by polygons, triangles and parallelograms (figure 7c&f).

2.3 *Rules for combining tonal blocks with lines*

lines	B	×	×	×	×	×	×	×	×	√	√	√	√	√	√	√	√
	W	×	×	×	×	√	√	√	√	×	×	×	×	√	√	√	√
block	B	×	×	√	√	×	×	√	√	×	×	√	√	×	×	√	√
	W	×	√	×	√	×	√	×	√	×	√	×	√	×	√	×	√
output		G	W	B	G	W	G	W	W	B	B	G	B	G	W	B	G

Table 1
Rules for determining the output tone (B=black, G=gray, W=white) to be rendered according to the tones of the lines and blocks (if present).

Having extracted the tonal blocks, a set of rules is required for determining how the various combinations and superpositions of lines and blocks of two tones plus gray background are rendered. This is given in table 1 which can be interpreted as drawing each line by its given tone unless there is a conflict with a line of the opposite tone or a conflict with a block. In the latter case, rendering the line takes precedence over the block. A black line over a white block can still be rendered black. However, a black line over a black block needs to be modified to be visible,

and is therefore rendered as gray. Inverting lines in this manner (i.e. modulated by their background) is a standard artistic technique, especially popular in woodblock illustration, and also used in previous artistic thresholding [17]. Perceptually more recognisable results are produced in our work than [17] by using one more tone.

The above rules are reasonable, but not the only possible choice. An example of commercial art that is close to our method is provided in figure 8 that demonstrates the combination of coloured regions and lines. Ignoring the red overlays, the poster also contains black and white blocks on a (yellow) background. Their rules cause overlaid blocks to modify colours of underlying blocks, e.g. for the amplifier white + white $\rightarrow$ black, white + black $\rightarrow$ yellow, etc. However, their scheme is more complex, since different rules are applied elsewhere, e.g. on the guitar white + white $\rightarrow$ yellow. Moreover, they include further rules for multiple intersections (e.g. the amplifier, guitar and white block: white + white + black $\rightarrow$ black, in which a double inversion has reverted to the original colour). Note that there are no special rules for modifying (e.g. inverting) overlaid lines.

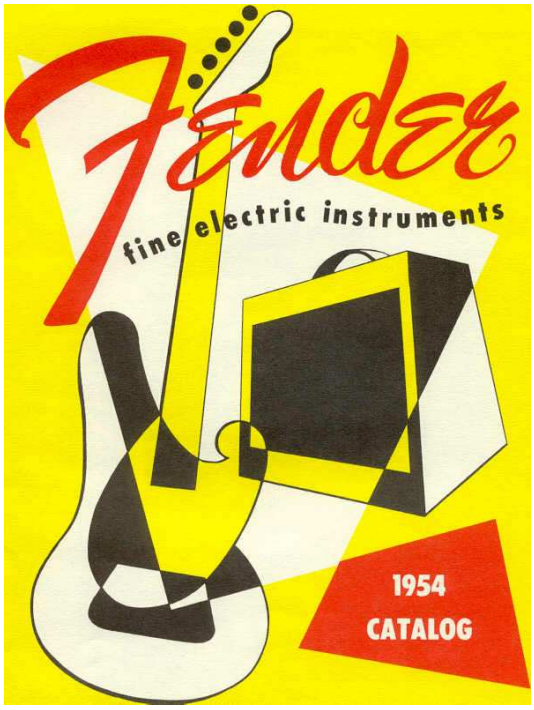

Fig. 8. An example from commercial graphic art of combining coloured regions and lines.

The effectiveness of our rules is demonstrated in figure 7. Even when the tonal block is distorted by the fitting of a parallelogram for the man's face in figure 7b the rules ensure that the line details are not obscured. A further good example is given by the rendering of the stripes on the right hand side of the man that is in the shadow.

Rather than render blocks as homogeneous we have experimented with incorporating a decorative pattern which is applied to (i.e. cut out from) each block. The

process of extracting the masks is illustrated in figure 6. Some examples using textured blocks are given in figures 1h, 18f and 25.

3 Extended Method

We now discuss some extensions or improvements to the basic method. Our method produces double responses for edges in the images which may not be ideal in some cases. We first look at possible approaches to suppress double responses while maintaining the visual quality of the output. Next, the problem of background clutter is addressed by performing foreground/background separation. We then consider how a multi-scale pyramid can be used to generate even richer results, at the cost of slightly more tones. Some variations of our basic method are then discussed, including coloured output and two-tone rendering. Many images contain human faces and more attractive rendering could be obtained with features such as human faces and eyes being detected and considered for emphasis or suppression.

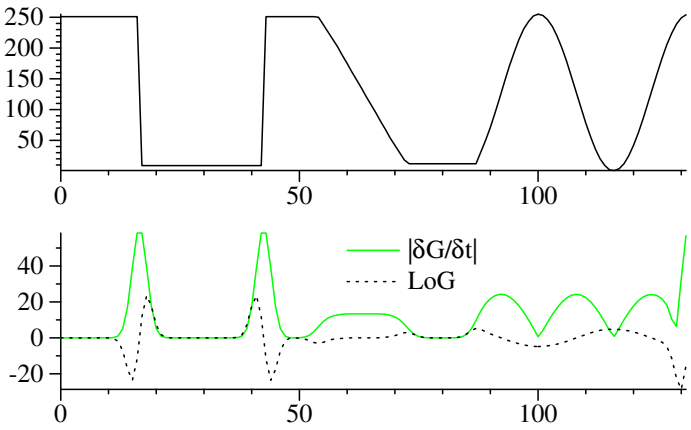

Fig. 9. A one dimensional signal (above), and responses of convolution by a LoG and $\frac{\partial G}{\partial t}$ (below).

3.1 Suppression of Double Responses from LoG

Kang *et al.* [10] used the DoG filter to extract lines. However, the DoG filter responds to edges as well as lines. For simplicity, we analyse the Laplacian of Gaussian (LoG) rather than the DoG. From the Gaussian

$$G(t,\sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{t^2}{2\sigma^2}}$$

the LoG is determined as

$$\nabla^2 G(t,\sigma) = \frac{\partial^2 G}{\partial t^2}(t,\sigma) = \frac{(t-\sigma)(t+\sigma)}{\sqrt{2\pi}\sigma^5} e^{-\frac{t^2}{2\sigma^2}}.$$

Modelling an ideal edge by the unit step function

$$U(t) = \begin{cases} 1 \text{ if } t \geq 0 \\ 0 \text{ if } t < 0 \end{cases}$$

the LoG response to an edge at the origin is given by the convolution

$$\nabla^2 G(t,\sigma) \otimes U(t) = -\frac{t}{\sqrt{2\pi}\sigma^3} e^{-\frac{t^2}{2\sigma^2}}.$$

Thus, while the LoG response is zero directly at the edge (leading to Marr and Hildreth's zero-crossing edge detector [36]), the extrema of $\nabla^2 G(t,\sigma) \otimes U(t)$ are symmetrically displaced about the edge, and are located at $t = \pm\sigma$, see figure 9.

Fig. 10. a) input image; b) three tone rendering; c) mask for edge suppression (white pixels suppressed); d) white strokes suppressed at edges; e) suppression using dilated mask; f) suppression using illumination heuristic.

The combined line and edge response is not problematic for Kang *et al.*, but is advantageous, since it ensures that the drawing captures additional salient features. However, in the context of our three tone drawing, not only are light and dark lines detected, but both of the LoG symmetric responses about the edges are retained (i.e. the LoG minima from the convolved image are extracted as LoG maxima from the convolved inverted image), producing a double response. This is evident in figure 10b along the strong edge of the collar.

One possible solution to the double response is to suppress one of the edge responses, e.g. the white strokes. If edge detection is applied, then the edge response can be compared against the LoG response to determine if the image window is more like an edge or line, and then suppress it in the former case. For edge detection we use the derivative of Gaussian

$$\frac{\partial G}{\partial t}(t,\sigma) = -\frac{t}{\sqrt{2\pi}\sigma^3} e^{-\frac{t^2}{2\sigma^2}}$$

which has the following edge response

$$\frac{\partial G}{\partial t}(t,\sigma) \otimes U(t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{t^2}{2\sigma^2}}.$$

At the location of the maximum of the LoG ($t = \pm\sigma$) the response of the derivative of Gaussian will be greater than the LoG so long as $\sigma > 1$. Figure 9 demonstrates how the derivative of Gaussian produces a higher response than the LoG at the edges, but a lower response at the valley at the right (which is matched in scale to the LoG).

When the derivative of Gaussian response is greater than the LoG response (during its application to the inverted intensity image for white stroke detection) edge suppression is performed by resetting the LoG to zero. However, the majority of the image is more edge like rather than line like (in figure 10c the line like pixels are coloured black), and the mask for retaining lines is sparse and thin. Consequently, since the image data does not perfectly conform to the line and edge models the line suppression tends to be excessive – see figure 10d. To counteract over-suppression the mask can be dilated; figure 10e shows reasonable results using a 3×3 circular structuring element.

A second approach to suppressing the double edge responses is based on the human visual system. It has been shown that when shading information is ambiguous humans tend to assume that the source of illumination is from above [37]. While this is a heuristic that is not universally applicable, many images will fall into this category, and the fact that humans follow this strategy justifies its effectiveness. Using this lighting assumption, a double edge response leads to the expectation that while a white line above a dark line will look natural, the reverse will not hold. This leads to the following simple heuristic: in the three tone image remove white lines that are directly below dark lines.

For every black pixel at (x, y), all white pixels at $(x, y + i); i = 1 \ldots D$ are reset to gray. Using a hysteresis approach, further white pixels are deleted: pixels $(x, y + i); i = (D+1) \ldots 2D$ are reset to gray if all the previous pixels visited in the second step, i.e. $(x, y + D)$ to $(x, y + i)$, were white before suppression.

Particularly in densely textured areas the results are improved by suppressing the line suppression, and so the process is applied in reverse to restore white lines above black lines. For every black pixel at (x, y), all deleted white pixels at $(x, y - i); i = 1 \ldots D$ are restored to white. The hysteresis approach is then applied to restore pixels in $(x, y - i); i = (D + 1) \ldots 2D$. The result of this approach is shown in figure 10f, where $D = 6$.

A third alternative is to perform no double response suppression. Not only does this avoid potential errors introduced in the two suppression methods, but it can be justified by its similarity to unsharp masking, which enhances images by emphasising edges.

3.2 Foreground/background separation

So far, we have applied rendering uniformly across the image. While this is often satisfactory, there will be some images in which it is necessary to modify the level of detail according to the saliency of objects in the image. Thus, two factors need to be considered: how to measure saliency, and a means of modifying the level of detail. Ideally these should be automatic, and for the first part there are many algorithms available. However, after experimentation we have not found any that are consistently reliable over a wide range of images (e.g. see figure 11). Therefore we take a three stage semi-automatic approach: the user "scribbles" in the image so as to very quickly and roughly indicate the whereabouts of the foreground and background. This is used to initialise a supervised watershed segmentation [38] in which colour models are built from the scribbles. As with the previously discussed segmentation of tonal blocks, such a global approach produces a good overall estimate of the foreground regions, but tends to have inaccurate boundaries (see figure 11c). Therefore we use grabcut again to refine the boundaries (see figure 11g).

Several approaches were investigated to modify the rendering based on the saliency map or foreground/background mask. The first two modified the input image before applying the basic three tone drawing method. Either the input image was blurred according to saliency, or else its contrast was reduced according to saliency. To modify the contrast the image's edge map was reduced according to saliency and then Poisson reconstruction [39] applied. The third method takes the basic three tone drawing generated from the input image and reduces the contrast of lines according to saliency. To avoid abrupt change in the output across the mask boundary, a small width of transition is added to smoothly interpolate between 0 and 1.

The need for figure/ground extraction is demonstrated in figure 11, which is a challenging example; the highly textured wall results in a profusion of background lines that swamp the figure of the boy in the foreground (figure 11h). Salience maps from [40–43] are displayed in figure 11a,b,e and f. While they are moderately effective at highlighting the boy they are too irregular and/or diffuse, and tend to miss parts. Consequently the rendered lines produced from the respective blurred input images are unsatisfactory (examples for the latter two are shown in figure 11i and 11j). We have tried further to refine the automatically extracted saliency map using turbo pixels [44], but the results are still not robust enough for general input images. In comparison, the scribble selected foreground mask clearly highlights the boy with only minor segmentation errors (figure 11g). The image is then blurred according to the mask (figure 11l), and the rendering successfully captures the boy with the wall still indicated but not dominating (figure 11k). For comparison, the alternative methods of using the foreground mask to modify the rendering are also shown. After the contrast of the background in the original image has been reduced using Poisson reconstruction (figure 11m) the basic three tone rendering in figure 11n is produced. It is very similar to the drawing from the blurred image (figure 11k) although the background lines are thinner. Figure 11o shows the back-

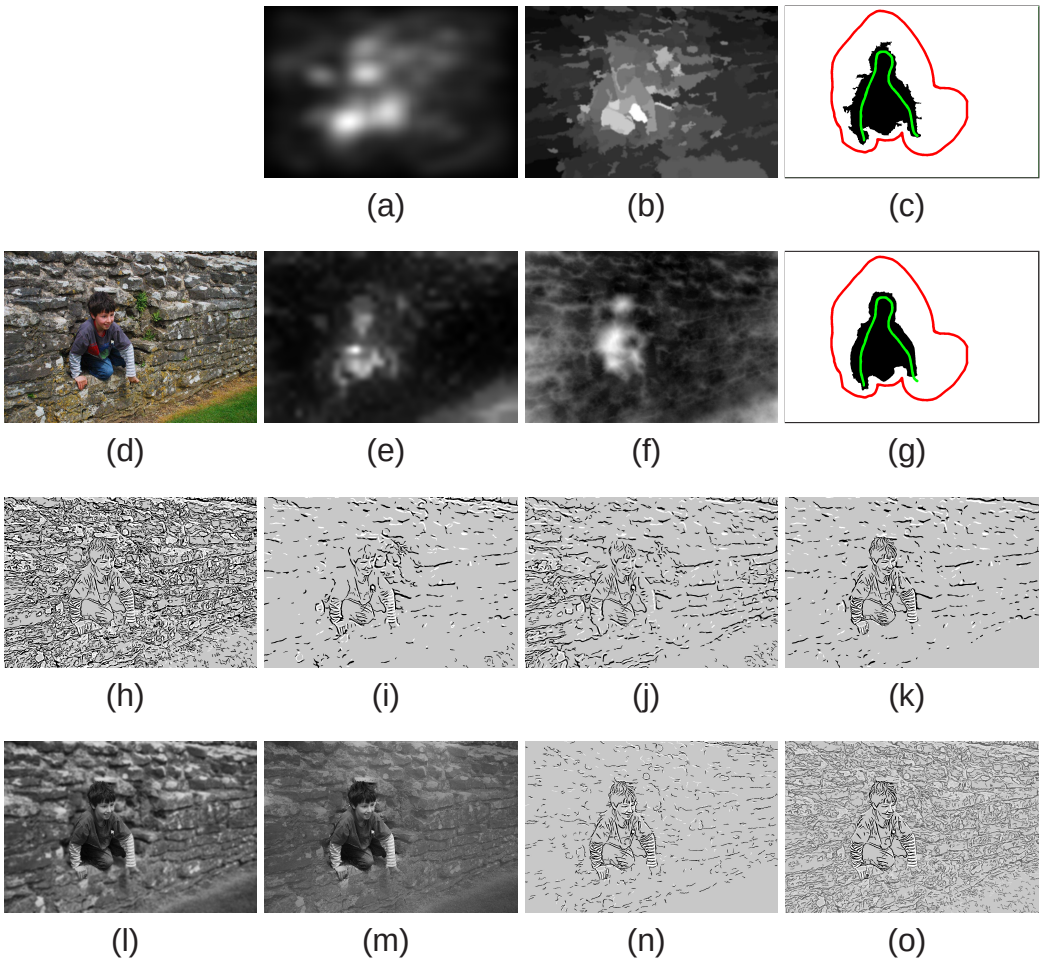

Fig. 11. Incorporating salience to modify level of detail: d) original image, a) salience map using method by [40], b) salience map using method by [41], e) salience map using method by [42], f) salience map using method by [43], c) mask after supervised watershed segmentation [38] showing initial scribbles for foreground and background, g) mask after refinement with Grabcut [33], h) our basic three tone rendering, i) three tone rendering after image is blurred using salience map from [42], j) three tone rendering after image is blurred using salience map from [43], k) three tone rendering after image is blurred using foreground mask, l) image blurred using foreground mask, m) background contrast reduced using Poisson reconstruction, n) three tone rendering from Poisson reconstruction, o) background lines from our basic three tone rendering rescaled (giving five tones).

ground lines rescaled; using the background mask the intensities of both the light and dark lines are mapped to 230 and 120 (while the midgray intensity is unaltered) to reduce their contrast. Although quite attractive (the boy is clearly highlighted) it is also more revealing of minor inaccuracies of the foreground mask.

3.3 Rendering pyramid

Earlier we showed that adding a third (gray) tone allowed the rendering to better capture an image's characteristics. Continuing this theme, we consider further increasing the number of tones, as a consequence of embedding the rendering in a multi-resolution framework. One approach for this would be to apply Kang *et al.*'s DoG detector with a range of kernel sizes to the input image. Instead, we choose to fix the kernel size, and apply the line detector to the image at different resolutions within a pyramid. The full three tone rendering method is applied at each resolution, and the outputs are rescaled to the full image size before being combined by simple averaging. Combination rules are thus applied to each level of resolution independently – this simple strategy avoids a combinatorial explosion in the number of rules. This approach also means that lines drawn at low resolution are made thicker after rescaling, which emphasises the salient lines. While resizing could be done using methods such as bilinear interpolation we use instead simple pixel replication followed by a small amount of Gaussian blurring which produces a smoother, more pleasing effect. Whichever of the two resizing methods is used, the interpolation or blurring will create new intensities that were not in the downsized image. Since we wish to retain a limited tonal palette, the intensities in the resized image are mapped onto the closest intensities in the downsized image.

How many tones does the mean image contain? Consider the general case of combining n images each containing the same t tones. The upper bound on the number of resulting mean tones will occur when each unique combination of tones produces a mean that is distinct from all other tonal combinations. The number of combinations of n items with repetition allowed from t items is $\binom{n+t-1}{n}$. For the example three level pyramids used in this paper the final rendering is still relatively minimal, containing $\leq \binom{5}{3} = 10$ tones.

Fig. 12. Rendering using the pyramid: a) original image, b) three tone rendering at bottom level, c) three tone rendering at middle level, d) three tone rendering at top level, e) mean of all renders from the pyramid.

Figure 12 demonstrates the application of the pyramid. Three tone renders from each of the three levels (without any morphological cleaning) are shown. The levels are combined into the mean image (with morphological cleaning, smoothing and intensity remapping) in figure 12e. Note the emphasis of significant lines in the final rendering by their detection and thickening at the higher levels of the pyramid.

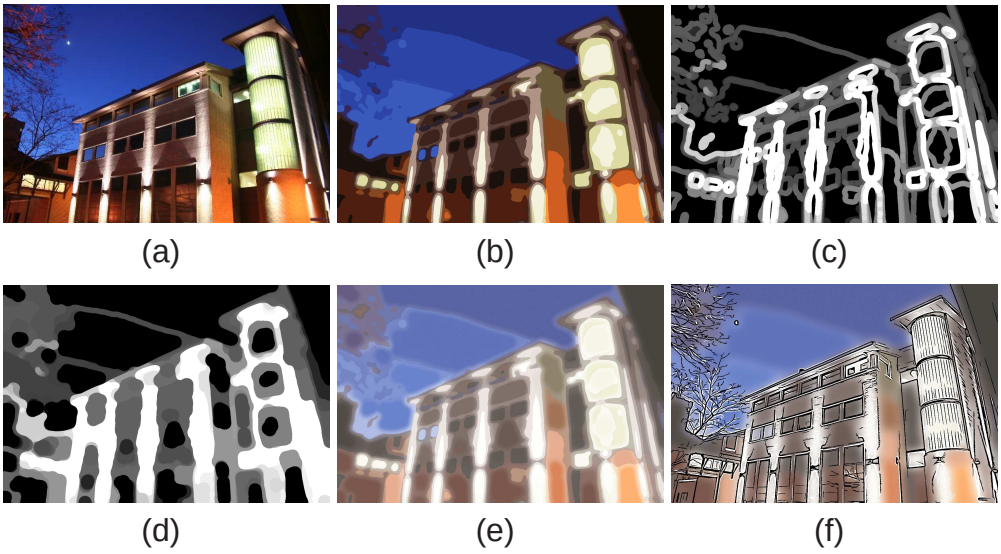

Fig. 13. Combining black and white lines with abstracted colour. a) original image, b) quantised colour map, c) colour map after dilation, d) after morphological opening, e) after variable intensity increase, f) final rendering with lines.

A modified version of our rendering pipeline has been produced that incorporates colour. The input image is first blurred, and the colours are quantised using K-means clustering of the CIE Lab histogram [45], which is a simpler approach than more sophisticated alternatives such as [46]. Next, since the number of clusters is affected by the number of histogram bins the process is run several times with different settings ($5, 10, \ldots 30$ bins in each dimension). The best colour clustering is selected by maximising the overlap between the boundaries in the quantised colour image ($\mathcal{C} = \{C_i\}$) and the black lines extracted from the original image ($\mathcal{L} = \{L_i\}$). Overlap is measured by computing

$$\frac{1}{|\mathcal{C}|}\sum_{C_i \in \mathcal{C}} \min_{L_j \in \mathcal{L}} ||C_i, L_j|| + \frac{1}{|\mathcal{L}|}\sum_{L_i \in \mathcal{L}} \min_{C_j \in \mathcal{C}} ||L_i, C_j||$$

where the minimum distances are efficiently calculated using the distance transform.

An optional second step can be applied to generate a diffuse colour effect. To this end, the colour regions are blurred, and their intensity is increased at their boundaries. To make the effect apply more strongly at high contrast boundaries the quantised colour image (figure 13b) is edge detected and the edges are dilated to produce a weighting map – see figure 13c. A morphological opening is further applied to simplify the weight map (figure 13d), followed by smoothing. After increasing the intensity according to the weight map, plus applying a further overall brightness increase to enhance the black lines (to be added in the last step), the result is shown in figure 13e. It can be seen that although the colours have been quantised such that

the sky contains several bands of blue, the low edge strength has ensured that there is only a small intensity change at their interface in the colour map (figure 13e), preventing an ugly artifact. This is further reduced by smoothing, after which the black and white lines are overlaid to produce the final result – see figure 13f.

While it might be expected that the distortion of the colour boundaries caused by the blurring for the diffuse effect would have an adverse effect, in fact this is not the case due to the presence of the lines. In the early days of poster design it was standard practise to use black outlines to cover the slight gaps and overlaps between regions of colour that occurred due to the imprecision of the printing process [47].

Combining simplified colour images with lines has been done by previous authors (e.g. [7,48]). However, they tend to just add black lines, and not white lines as we have done.

3.5 Two-tone drawings

A different modification of our rendering pipeline dispenses with all intermediate tones and colours, and generates a two tone rendering. The dark tonal blocks are generated as before except that two threshold classes are used instead of four, and then the dark and light lines are simply directly overlaid (without applying the special combination rules). See figures 17c, 18c and 19c for examples.

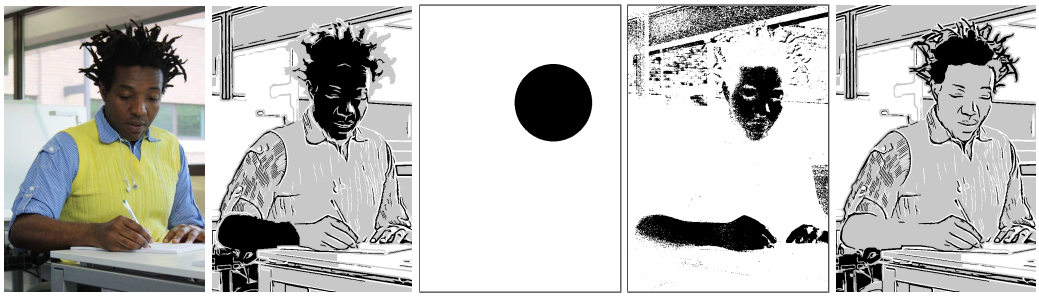

Fig. 14. Improving rendering using face detection and tonal block suppression. a) original image, b) three tone rendering without face detection, c) detected face mask, d) detected skin pixels (black), e) our rendering (3 tones).

3.6 Face processing

Our basic rendering pipeline was made available online (`http://www.cs.cf.ac.uk/npr/`) and user feedback indicated that a recurring failing occurred when the tonal blocks covered the face. Even though the combination rules ensured that the facial details were still fully captured, the sensitivity of the human visual system to the face meant that examples such as those in figure 14b were considered unattractive. As

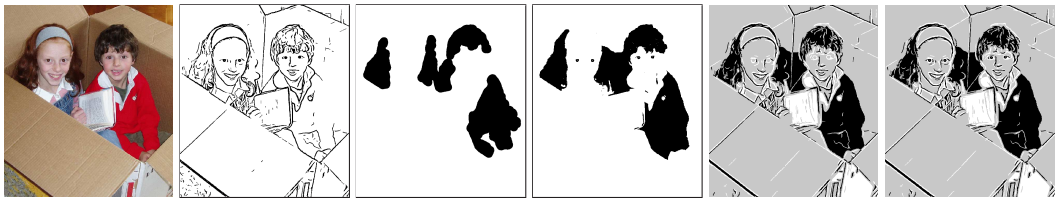

Fig. 15. Improving rendering using eye detection. a) original image, b) dark lines c) dark mask after cleaning by mathematical morphology, d) dark mask after grabcut, e) our rendering, f) final rendering with eye detail restored.

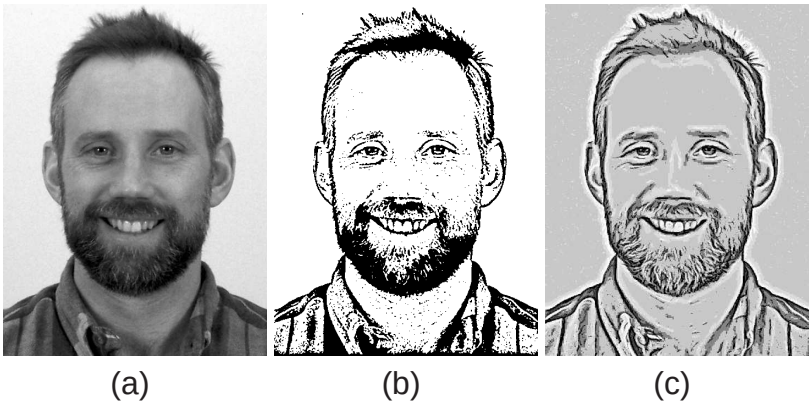

Fig. 16. a) original image, b) rendering from [29], c) our 10 tone rendering.

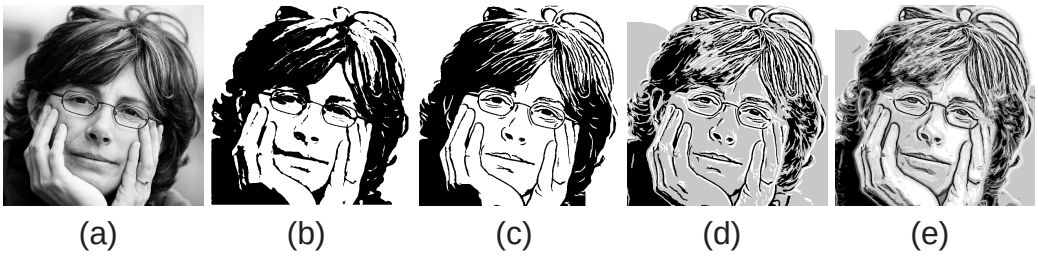

Fig. 17. Comparison with Mould and Grant. a) original image, b) rendering from [16], c,d,e) our 2, 3, 10 tone renderings.

human faces appear very often in many images, we use face detection to (optionally) suppress tonal blocks on faces. Faces are first detected with a fast object detection algorithm using boosted cascading classifiers with Haar-like features [49], as shown in figure 14c. The output of the face detector as shown in this example is not very accurate. This is sufficient for many applications, but for the purpose of rendering, inaccuracy will incur artefacts. To improve the accuracy and robustness, we also use skin detection [50] to find those skin-like pixels as shown in figure 14d. Only those pixels belonging to both masks are recognised as part of the face, and those pixels are used to build a more accurate image specific skin model. Those refined skin pixels will be removed from the tonal blocks before running grabcut, and are also excluded from grabcut optimisation to prevent them being included again. The result with face and skin tonal block suppression is given in figure 14e.

Fig. 18. Comparison with Xu and Kaplan. a) original image, b) rendering from [17], c,d) our 2, 3 tone renderings, e) our rendering (triangles fitted to tonal blocks) f) our rendering (textured tonal blocks).

Fig. 19. Comparison with Xu and Kaplan. a) original image, b) rendering from [17], c,d) our 2, 3 tone renderings.

A second problem that is perceptually important occurs when the eyes are poorly drawn. In figure 15 we see that although the basic lines capture the eyes, the final refined dark tonal block also includes the eyes. The combination rule for dark line overlayed on dark tonal block means that the resulting output is gray, causing some of the eye detail to be lost (see figure 15e). However, the eye detection system can be used to provide a mask specifying that in that area of the rendered image the (black and white) lines are directly overlayed (without using the combination rules). In addition, these lines have a reduced amount of morphological filtering applied (the size threshold is halved) since it is preferable to potentially include some unnecessary clutter in the eye region if it ensures that the necessary eye detail is more likely to be retained. Eye detection can also be used to produce some artistic effects. For example, figure 1i combines direct cutout of eyes with rendering of the remaining image (and thus has the eyes emphasised).

Fig. 20. Comparison with Orzan *et al.* a) original image, b) rendering from [48], c) our rendering (3 tones) d) our rendering (lines and colour).

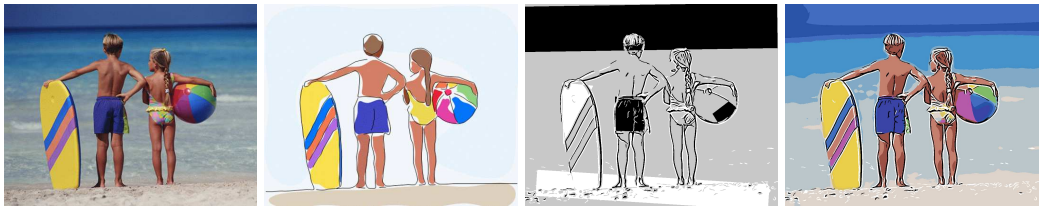

Fig. 21. Comparison with Wen *et al.* a) original image, b) rendering from [11], c) our rendering (3 tones) d) our rendering (lines and colour).

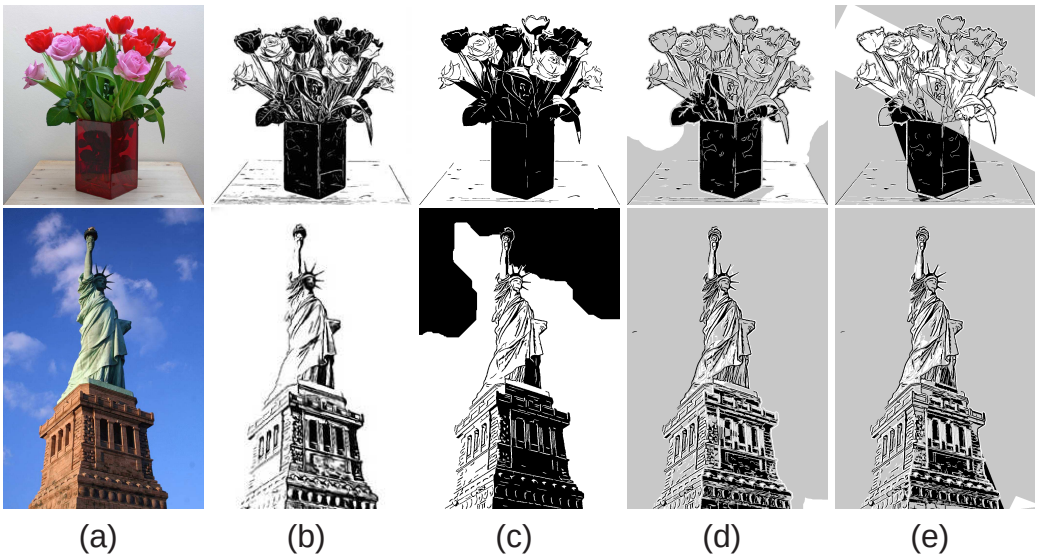

Fig. 22. Comparison with Winnemöller *et al.* a) original image, b) rendering from [19], c) our rendering (2 tones) d) our rendering (3 tones), e) our rendering (region simplification).

4 Results

For simplicity the rendering algorithm has assumed a standard image size of about 1 megapixel (the examples in this paper were mostly 1 megapixel $\pm 20\%$), and the basic algorithm's parameter values (which were used for all the results in the paper) have been set as: for hysteresis thresholding $T_U = 200$ and $T_L = 100$; if the noise level is > 2 then the size value for area based opening and closing is 16; before grabcut refinement the tonal blocks are simplified using area based closing and opening with sizes 9000 and 1000 respectively, followed by an opening with a circular structuring element of diameter 31; the final filtering of the rendering uses an area based opening and closing of size 32.

Programs were run on a 2.40 GHz Intel Core 2 Duo with 4GB of RAM. For 1 megapixel images the run time of our basic method was about 20 seconds using our unoptimised code.

Figure 16 shows an example from Gooch *et al.* [29] where their drawing is generated by performing a logical AND of the thresholded raw and line detected images.

Their thresholds are carefully tuned, and they are able to achieve high quality results. Our rendering using a three level pyramid also captures the subject's resemblance from the input photograph, but the extra tones enable detailing (such as the beard) and subtleties (such as the bridge of the nose) to be retained.

Figure 17 shows an example from Mould and Grant [16] who combine a base layer (computed using graph cuts) with a detail layer (computed by local thresholding). Their results are attractive, but have a photographic appearance to them, as opposed to our renderings which look more hand drawn. In addition, even in just our two tone and three tone versions, features such as the eyes and hands have been more clearly delineated.

Figure 18 shows an example from Xu and Kaplan [17] who oversegment the colour image and then optimise the assigned segment tones (black/white) according to their intensity, boundary contrast, etc. Thus segments are sometimes inverted from their natural tone so as to maintain boundaries (cf. our rules for overlaying lines and blocks). However, compared to our results, their stylisation destroys or distorts a lot of important detail (such as the eyes). Figure 18e shows how we can increase the degree of abstraction (by fitting only triangles to the tonal blocks) but still retain detail using the lines. A second example from [17] is given in figure 19. To ensure a contrast between the tree and both the building and the sky, Xu and Kaplan introduce a feature crossing through the tree, and obtain the result in figure 19b. Our three tone version adequately represents tree, building and sky; the sky is correctly rendered as a white tonal block, although due to its relatively thin branching structure the tree is not detected as a black tonal block (it is removed by the morphological cleaning step). All three objects are also rendered well by our two tone version – the white lines ensuring that the tree is separated from the background.

Figure 20 shows an example from Orzan *et al.* [48] who extract edges at multiple scales, and then reconstruct the image from the salient edges (those with large lifetime) by solving a Poisson equation. In addition, in figure 20b they overlay a rendering of the edges with their width proportional to their lifetime. Our colour rendering (figure 20d) is qualitatively similar, although note that the line details are rather clearer.

Figure 21 shows an example from Wen *et al.* [11] who perform interactive segmentation, region simplification and shrinking, and colour modification. Our automatic result in figure 21c fits the geometric models with the exclusion of polygons to increase the abstraction (e.g. causing the sand at the bottom to be represented by the rectangle).

Figure 22 compares the XDoG method with ours. Both methods are able to generate somewhat similar 2 tone results with inverted lines – this is demonstrated in the second and third columns. Note that while our results are strictly two tone the XDoG settings used here produced a soft thresholding effect. In addition, both methods are also capable of producing other styles of their own. The results of our method which cannot be directly achieved by XDoG are shown in the fourth and

fifth columns (3 tone without and with region simplification). The former preserves the tonal balance of the original image, while the latter demonstrates one of our stylisation effects. Using triangular abstraction for the statue's base produces a relatively subtle effect, whereas abstraction using rectangles for the vase and flowers is more dramatic.

Figure 23 shows an example which highlights the limitations of the current method for extracting tonal blocks. Since the thresholding algorithm is applied to the gray level version of the image it is unable to extract meaningful tonal blocks (figure 23c). If colour blocks as described in section 3.4 are used instead then satisfactory results are obtained (figure 23d).

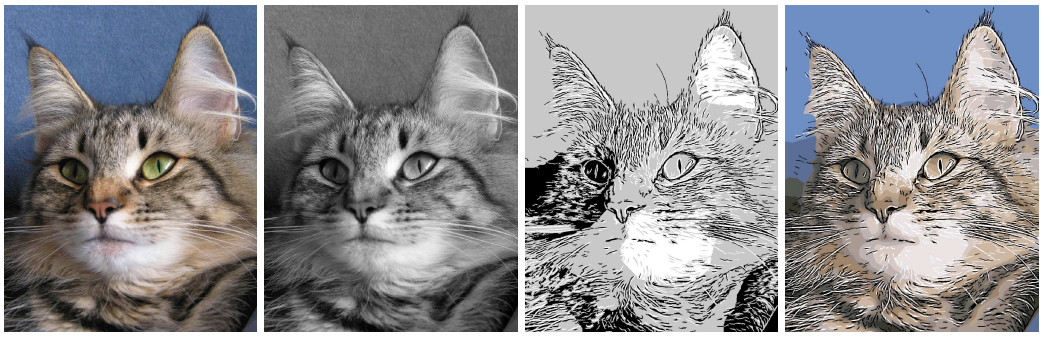

Fig. 23. Problems with tonal blocks. a) original image, b) gray level version of image used for thresholding, c) 3 tone rendering, d) line and coloured region rendering.

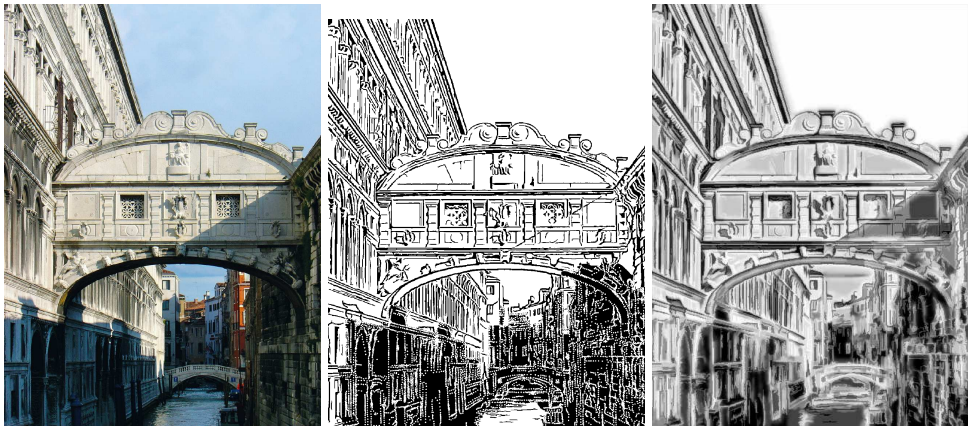

Fig. 24. Alternative rendering styles. a) original image, b) dotted lines, c) blurred pyramid.

There are many other modifications of our system pipeline that can be applied to produce alternative artistic effects. For instance, in figure 24b the method is inspired by Aubrey Beardsley's pen and ink drawings that sometimes use a series of dots in place of lines. In our algorithm, the black lines overlaid with black tonal blocks are drawn as white dots. Since the lines are pixel images rather than strokes, the dots are generated by randomly sampling points in the line mask and then iterating the centroidal Voronoi diagram to produce a more regular stippling [51].

In figure 24c a more diffuse effect is generated compared to our normal pyramid based result by blurring the result pyramid, with the degree of blurring increasing at higher levels of the pyramid. In the example, the result at the scale of the input image is not blurred, at the next level Gaussian smoothing with $\sigma = 2$ is used, and at the next (top) level Gaussian smoothing with $\sigma = 4$ is used. Thus, although substantial blurring is applied which creates the diffuse effect, the lower levels of the pyramid retain sufficient detail to enable easy recognition of the image.

Figure 25 and figure 26 show various artistic rendering results. To provide a diverse collection of results we choose among the combinations of two tone or three tone, single level or pyramid-based, simplified or non-simplified blocks, textured or untextured blocks. As we noted before, the combination of lines and tonal blocks is capable of providing very effective renderings of faces [27], and this is further demonstrated in figure 27.

As the gallery shows, our pipeline works effectively on a wide variety of images. All the components in the pipeline are robust apart from the tonal block extraction, as discussed above. To obtain good results the input image has to satisfy certain requirements. First, it should have reasonably high contrast and resolution, otherwise desirable lines may be lost. Second, if the scene is too cluttered this will result in a rendering with visual clutter, although foreground/background separation can be used to alleviate this.

Finally, our line and tonal block rendering method captures the salient information in the image, and as such is useful for generating reduced resolution versions (i.e. thumbnails) of images. To enhance identity recognition we combine the three tone rendering (in which a lower threshold on morphological cleaning is required since the input image is so small) with the gray scale image by taking their average, and then further overlaying the black (white) tonal blocks by a logical AND (OR respectively). Experiments were made in which the black and white lines were logically combined in the same manner, but at such low resolution the lines were too thick (obliterating neighbouring detail) or too fragmented, and the results were unsatisfactory. Figure 28 shows the results of reducing the images to 48×54. Although in some cases the tonal blocks have failed to capture all the relevant dark areas in the image, the majority of important features are retained, and the lines have succeeded in enhancing the identity of the portraits.

4.1 User Study

It is a major challenge to evaluate the effectiveness of non-photorealistic rendering algorithms [52–54]. While aesthetic quality is largely subjective, the effectiveness of abstraction in non-photorealistic rendering can be measured more objectively. Thus we limit our user study to evaluate the effectiveness of abstraction only, i.e. to verify that our renderings preserve or even emphasise perceptually important features from the images. Previous work used the tasks of face recognition [29]

and a memory tile game [8] to demonstrate the advantages of their renderings over photographs. We adopt the memory tile game paradigm using faces for two reasons: it is more culturally neutral as compared to remembering names of people in face recognition tasks, and more fun as a game to potentially attract more volunteers. Face discrimination is a demanding task, requiring differentiation of subtle features, and so it is reasonable to use faces as a representative dataset and to expect that the findings will generalise to a wider image content (as done by [29,8]).

Participants We implemented our user study as a web-based game. 57 volunteers (25% female, ages 18-50) were involved, including undergraduates, post-graduate students and staff.

Material Unlike previous studies, we are more interested to see how our non-photorealistic rendering compared with other approaches, especially a human artist's drawings. We use the full 150 faces in the first disc of the XM2VTS multi-modal face database [55] and the artist's drawings from the CUHK Face Sketch Database [56], drawn by an artist with professional skills, based on the XM2VTS photos. [2] We also generated results with typical renderings described in this paper, namely our 3-tone method and multi-scale pyramid (10-tone) version, as well as a standard mosaic style rendering [57] to provide a baseline; see figure 29 for some examples. The input images from the database are at 720×576 resolution. All images were first normalised into a standardised coordinate frame by translation and scaling using the manually marked coordinates of the eyes (provided also from the database). We used the same fixed settings as described in this section to produce our renderings, fully automatically. The images were then downsized to 200×250 resolution. This allows a reasonable grid of images to be displayed at the same time on a normal computer display. For our renderings, we used nearest neighbour interpolation (i.e. dropping out certain columns or rows of pixels) instead of say bicubic interpolation to avoid introducing more tones.

Procedure The memory tile game basically shows a 3×6 grid of cards with back side up and every time the player clicks a pair of cards they will be revealed for a short time (0.5 seconds in our experiments). If the cards uncovered by two adjacent clicks are matched then both cards are removed, otherwise they are reset to their previous state. Different from a traditional memory tile game, we asked the player to match grayscale photos with one of the four non-photorealistic rendering styles (including the human artist's sketches). The time and number of clicks needed to complete the game are recorded. Players were first asked to play a trial version to familiarise themselves with the system followed by a testing mode where timings were recorded. Each player was asked to do all the four tasks with different non-photorealistic styles in a random order to avoid bias to expertise (which might favour later tasks) or concentration (which might favour earlier tasks).

[2] Note that although some of the artist's drawings show small distortions and pose change with respect to the original photographs, the portraits all preserve the true likenesses of the individuals.

Results Table 2 shows the statistics of our experiments. On average our 3-tone rendering achieves the minimal average time and number of clicks, performing even better than the human artist's sketches. Our multi-scale pyramid renderings achieve similar performance with the artist's sketches (slightly greater average time and slightly lower number of clicks). The mosaic results perform worst for both average time and number of clicks. It is not surprising as this rendering style loses many significant features potentially helpful for memorising and recognition. We use two-way ANOVA (Analysis of Variance) to analyse the data with the null hypothesis being no difference between two styles in this task, and $\alpha = 0.01$. For this task, our 3-tone drawing is comparable to the artist's sketches, with a reduced average time and number of clicks ($p = 0.2145$ for time, and $p = 0.2646$ for number of clicks). Moreover, it is better with statistical significance ($p = 0.0018$ for time and $p = 0.0060$ for number of clicks) than mosaics while the other styles (even the artist's sketches) are not statistically significantly better than mosaics.

Discussion Our experiment asked participants to match photos with non-photorealistic drawings. Our 3-tone rendering, which contains less detail than several of the alternative renderings, worked better on average for both times and number of clicks than our multi-scale pyramid rendering as well as artist's drawings, both of which contain more detail (see figure 29 for examples). This shows that our minimal rendering is effective and beneficial for the task of memorising since it captures and retains salient features whilst discarding non essential features.

	3-tone	Pyramid	Artist	Mosaic
Average Time (sec)	45.19	48.68	47.75	53.53
p	0.0018	0.1297	0.0393	–
F	10.70	2.37	4.46	–
Average Click Number	51.19	53.75	53.96	58.25
p	0.0060	0.1490	0.1217	–
F	8.16	2.14	2.47	–

Table 2
Statistics of user study experiments. All p and F values are calculated for the methods versus the mosaics. The degrees of freedom for all tests are one.

5 Conclusion

In this paper, aiming towards producing a minimal rendering of images, we propose a new algorithm to generate artistic renderings. Lines and blocks are extracted, simplified and appropriately combined to produce visual pleasing non-photorealistic rendering results. Combination of lines and colour is commonplace and generates attractive results (both in previous work and our results presented here). However,

tackling the minimal approach, and using just three (or a small number of) tones, is less straightforward. Our combination rules ensure that tonal blocks can be included to provide (sometimes essential) overall tonal balance, and details provided by the lines are also well retained. As there is no unique answer for artistic rendering, a variety of rendering styles (such as colour, 2-tone etc.) have been explored. All the examples presented in this paper are obtained fully automatically. Our algorithm generally works well for a variety of types of images, as demonstrated in the paper. For example, for images of faces its effectiveness at capturing the salient characteristics was verified by a careful user study.

The inclusion of dark and light tonal blocks greatly enhances the appearance of the renderings (as demonstrated by various examples throughout the paper). However, even with our two stage thresholding and grabcut approach it is not possible to ensure that no errors occur. Various techniques have been used in this paper to reduce these effects. The first is to replace the blocks by highly abstracted versions that simultaneously hides the faulty segmentation and provides an attractive stylisation. A BIC-based method is used to automatically determine the most appropriate abstraction balancing the complexity and accuracy of elementary shapes. The second is to use a multi-resolution framework so that a faulty segmentation at one level will often be partially rectified by the results at another level. Some rendering styles such as the colour diffusion style also help to hide inaccuracy in the tonal blocks.

A few further directions can potentially be explored in the future. As mentioned, our user study focuses on the abstraction aspect of rendering (i.e. representing features significant for recognition). In the future it would be interesting to evaluate the aesthetics of non-photorealistic renderings, but this is known to be difficult due to both the problem of quantifying aesthetics as well as its subjective nature [52]. Our current rendering process is primarily bottom up. Likewise, investigating the effect of varying the "minimality" of the rendering, while of interest, is complicated by the need for quantifying minimality; standard computational schemes, e.g. using entropy, are unlikely to correspond to perceptual judgments. A future goal is to incorporate semantic information since this would enable us to more closely approach the goal of minimal rendering, and a learning based approach using prior knowledge may be employed; similar ideas have been used for determining semantically meaningful line drawings from 3D shapes [58].

Acknowledgements

We would like to thank Henry Kang for providing code for his line drawing algorithm, Michael Wilkinson for providing his connected set morphology code, Marco Eichmann and Martin Luessi for providing their multilevel image thresholding code, Justin Talbot for providing his GrabCut code, and Yi-Zhe Song for providing figure 1c.

References

[1] A. Hausner, Simulating decorative mosaics, in: Proc. ACM SIGGRAPH, 2001, pp. 573–580.

[2] D. Mould, A stained glass image filter, in: Eurographics Workshop on Rendering Techniques, 2003, pp. 20–25.

[3] A. Hertzmann, Painterly rendering with curved brush strokes of multiple sizes, in: Proc. ACM SIGGRAPH, 1998, pp. 453–460.

[4] S. Hiller, H. Hellwig, O. Deussen, Beyond stippling – methods for distributing objects on the plane, Comput. Graph. Forum 22 (3) (2003) 515–522.

[5] C.-K. Yang, H.-L. Yang, Realization of Seurat's pointillism via non-photorealistic rendering, The Visual Computer 24 (5) (2008) 303–322.

[6] M. Zhao, S. C. Zhu, Sisley the abstract painter., in: ACM Symp. NPAR, 2010, pp. 99–107.

[7] J. Kyprianidis, J. Döllner, Image abstraction by structure adaptive filtering, in: Proc. EG UK Theory and Practice of Computer Graphics, 2008, pp. 51–58.

[8] H. Winnemöller, S. Olsen, B. Gooch, Real-time video abstraction, ACM Trans. Graphics 25 (3) (2006) 1221–1226.

[9] Y. Song, P. Hall, P. L. Rosin, J. Collomosse, Arty shapes, in: Proc. Comp. Aesthetics, 2008, pp. 65–73.

[10] H. Kang, S. Lee, C. K. Chui, Coherent line drawing, in: ACM Symp. NPAR, 2007, pp. 43–50.

[11] F. Wen, Q. Luan, L. Liang, Y.-Q. Xu, H.-Y. Shum, Color sketch generation, in: ACM Symp. NPAR, 2006, pp. 47–54.

[12] D. DeCarlo, S. Rusinkiewicz, Highlight lines for conveying shape, in: ACM Symp. NPAR, 2007, pp. 63–70.

[13] Y. Lee, L. Markosian, S. Lee, J. Hughes, Line drawings via abstracted shading, ACM Trans. Graphics 26 (3) (2007) 18:1–5.

[14] P. L. Rosin, Y.-K. Lai, Towards artistic minimal rendering, in: ACM Symp. NPAR, 2010, pp. 119–127.

[15] D. Duke, I. Herman, Minimal graphics, IEEE Computer Graphics & Application 21 (6) (2001) 18–21.

[16] D. Mould, K. Grant, Stylized black and white images from photographs, in: ACM Symp. NPAR, 2008, pp. 49–58.

[17] J. Xu, C. S. Kaplan, Artistic thresholding, in: ACM Symp. NPAR, 2008, pp. 39–47.

[18] M. Meng, M. Zhao, S. C. Zhu, Artistic paper-cut of human portraits, in: 18th Int. Conf. on Multimedia, 2010, pp. 931–934.

[19] H. Winnemöller, J. E. Kyprianidis, S. C. Olsen, XDoG: An eXtended difference-of-gaussians compendium including advanced image stylization, Computers & Graphics 36 (6) (2012) 740–753.

[20] B. Buchholz, T. Boubekeur, D. DeCarlo, M. Alexa, Binary shading using appearance and geometry., Comput. Graph. Forum 29 (6) (2010) 1981–1992.

[21] J. Bronson, P. Rheingans, M. Olano, Semi-automatic stencil creation through error minimization, in: ACM Symp. NPAR, 2008, pp. 31–37.

[22] V. Rivotti, J. Proẹnca, J. Jorge, M. Sousa, Composition principles for quality depiction and aesthetics, in: Proc. Comp. Aesthetics in Graphics, Visualization, and Imaging, 2007, pp. 37–44.

[23] J. Immerkaer, Fast noise estimation, Computer Vision and Image Understanding 64 (1996) 300–302.

[24] J. Canny, A computational approach to edge detection, IEEE Trans. PAMI 8 (1986) 679–698.

[25] A. Meijster, M. Wilkinson, A comparison of algorithms for connected set openings and closings, IEEE Trans. PAMI 24 (4) (2002) 484–494.

[26] M. Luessi, M. Eichmann, G. Schuster, A. Katsaggelos, Framework for efficient optimal multilevel image thresholding, J. of Electronic Imaging 18 (2009) 013004+.

[27] V. Bruce, E. Hanna, N. Dench, P. Healey, M. Burton, The importance of mass in line drawings of faces, Applied Cognitive Psychology 6 (7) (1992) 619–628.

[28] D. Pearson, J. Robinson, Visual communication at very low data rates, Proc. IEEE 73 (4) (1985) 795–812.

[29] B. Gooch, E. Reinhard, A. Gooch, Human facial illustrations: Creation and psychophysical evaluation, ACM Trans. Graphics 23 (1) (2004) 27–44.

[30] M. Sezgin, B. Sankur, Survey over image thresholding techniques and quantitative performance evaluation, J. of Electronic Imaging 13 (1) (2004) 146–168.

[31] N. Otsu, A threshold selection method from gray-level histograms, IEEE Transactions on Systems, Man, and Cybernetics 9 (1979) 62–66.

[32] W. Tao, H. Jin, Y. Zhang, Color image segmentation based on mean shift and normalized cuts, IEEE Transactions on Systems, Man, and Cybernetics, Part B 37 (5) (2007) 1382–1389.

[33] C. Rother, V. Kolmogorov, A. Blake, "GrabCut": interactive foreground extraction using iterated graph cuts, ACM Trans. Graphics 23 (3) (2004) 309–314.

[34] K. Voss, H. Süße, Invariant fitting of planar objects by primitives, IEEE Trans. PAMI 19 (1) (1997) 80–84.

[35] U. Ramer, An iterative procedure for the polygonal approximation of plane curves, Computer, Graphics and Image Processing 1 (1972) 244–256.

[36] D. Marr, E. Hildreth, Theory of edge detection, Proceedings of the Royal Society, London, B 207 (1980) 187–217.

[37] J. O'Shea, M. Agrawala, M. Banks, The preferred angle of illumination in shape from shading, J. Vision 8 (6) (2008) 444:1–8.

[38] S. Lefèvre, Knowledge from markers in watershed segmentation, in: Computer Analysis of Images and Patterns, Vol. 4673 of LNCS, 2007, pp. 579–586.

[39] P. Pérez, M. Gangnet, A. Blake, Poisson image editing, ACM Trans. Graphics 22 (3) (2003) 313–318.

[40] J. Li, M. D. Levine, X. An, X. Xu, H. He, Visual saliency based on scale-space analysis in the frequency domain, IEEE Transactions on Pattern Analysis and Machine Intelligence.

[41] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, S.-M. Hu, Global contrast based salient region detection, in: CVPR, 2011, pp. 409–416.

[42] J. Harel, C. Koch, P. Perona, Graph-based visual saliency, in: NIPS, 2006, pp. 545–552.

[43] P. L. Rosin, A simple method for detecting salient regions, Pattern Recognition 42 (11) (2009) 2363–2371.

[44] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, K. Siddiqi, TurboPixels: Fast superpixels using geometric flows, IEEE Trans. PAMI 31 (12) (2009) 2290–2297.

[45] T. Ohashi, Z. Aghbari, A. Makinouchi, Hill-climbing algorithm for efficient color-based image segmentation, in: Proc. IASTED, 2003, pp. 17–22.

[46] N. Papamarkos, A. Atsalakis, C. Strouthopoulos, Adaptive color reduction, IEEE Transactions on Systems, Man, and Cybernetics, Part B 32 (1) (2002) 44–56.

[47] W. Raffé, Poster Design, Chapman and Hall, 1929.

[48] A. Orzan, A. Bousseau, P. Barla, J. Thollot, Structure-preserving manipulation of photographs, in: ACM Symp. NPAR, 2007, pp. 103–110.

[49] P. Viola, M. J. Jones, Rapid object detection using a boosted cascade of simple features, in: Proc. CVPR, 2001, pp. Vol. I: 511–518.

[50] R. L. Hsu, M. Abdel-Mottaleb, A. Jain, Face detection in color images, IEEE Trans. PAMI 24 (5) (2002) 696–706.

[51] A. Secord, Weighted voronoi stippling, in: ACM Symp. NPAR, 2002, pp. 37–43.

[52] A. Hertzmann, Non-photorealistic rendering and the science of art, in: ACM Symp. NPAR, 2010, pp. 147–157.

[53] N. Redmond, J. Dingliana, Investigating the effect of real-time stylisation techniques on user task performance, in: Proc. Symposium on Applied Perception in Graphics and Visualization, 2009, pp. 121–124.

[54] A. Santella, D. DeCarlo, Visual interest and npr: an evaluation and manifesto, in: ACM Symp. NPAR, 2004, pp. 71–150.

[55] K. Messer, J. Matas, J. Kittler, J. Luettin, G. Maitre, XM2VTSDB: the extended of M2VTS database, in: Proc. Audio- and Video-Based Person Authentication, 1999, pp. 72–77.

[56] X. Wang, X. Tang, Face photo-sketch synthesis and recognition, IEEE Trans. PAMI 31 (11) (2009) 1955–1967.

[57] P. Haeberli, Paint by numbers: abstract image representations, in: SIGGRAPH, 1990, pp. 207–214.

[58] F. Cole, A. Golovinskiy, A. Limpaecher, H. S. Barros, A. Finkelstein, T. Funkhouser, S. Rusinkiewicz, Where do people draw lines?, ACM Trans. Graphics 27 (3) (2008) 88:1–11.

Fig. 25. Three tone artistic renderings with various styles produced with our algorithm.

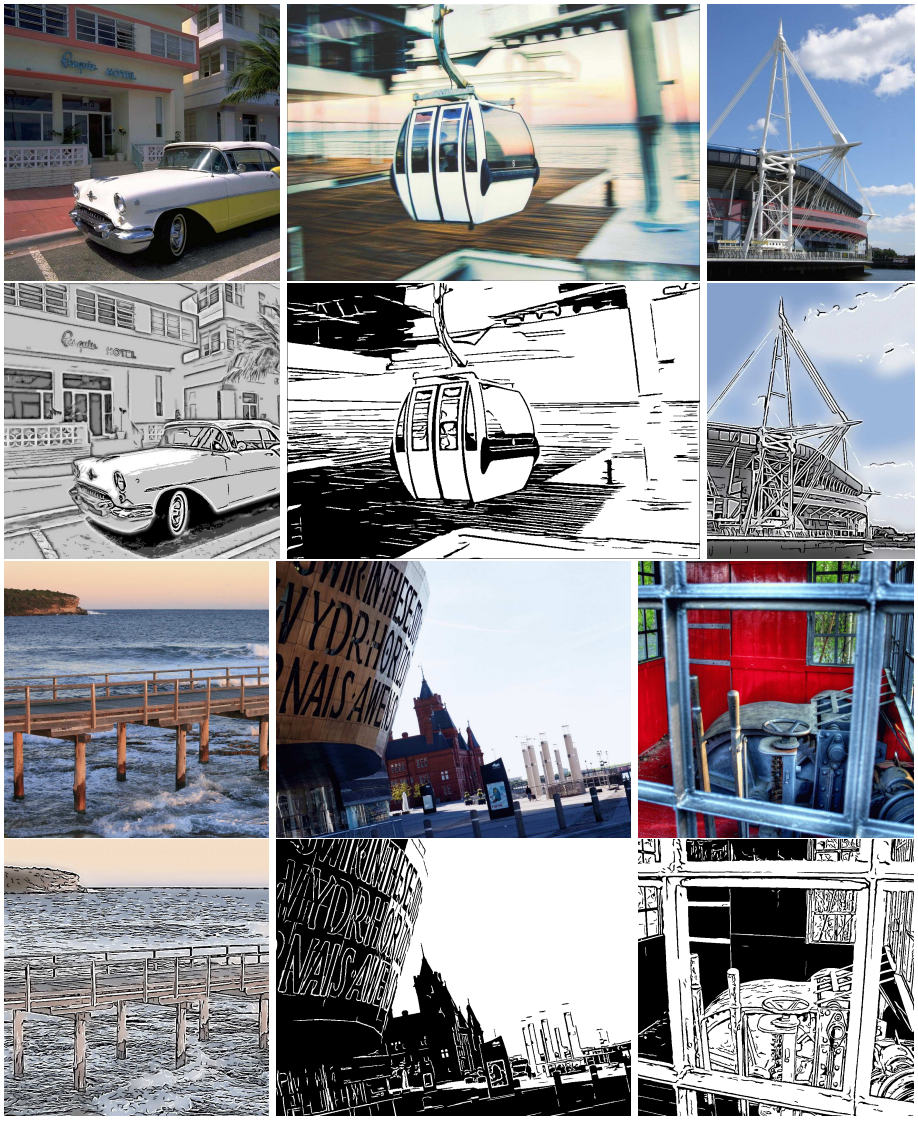

Fig. 26. Artistic renderings with two tone, 10 tone and colour produced with our algorithm.

Fig. 27. Two tone renderings of famous faces.

Fig. 28. Thumbnail renderings of famous faces.

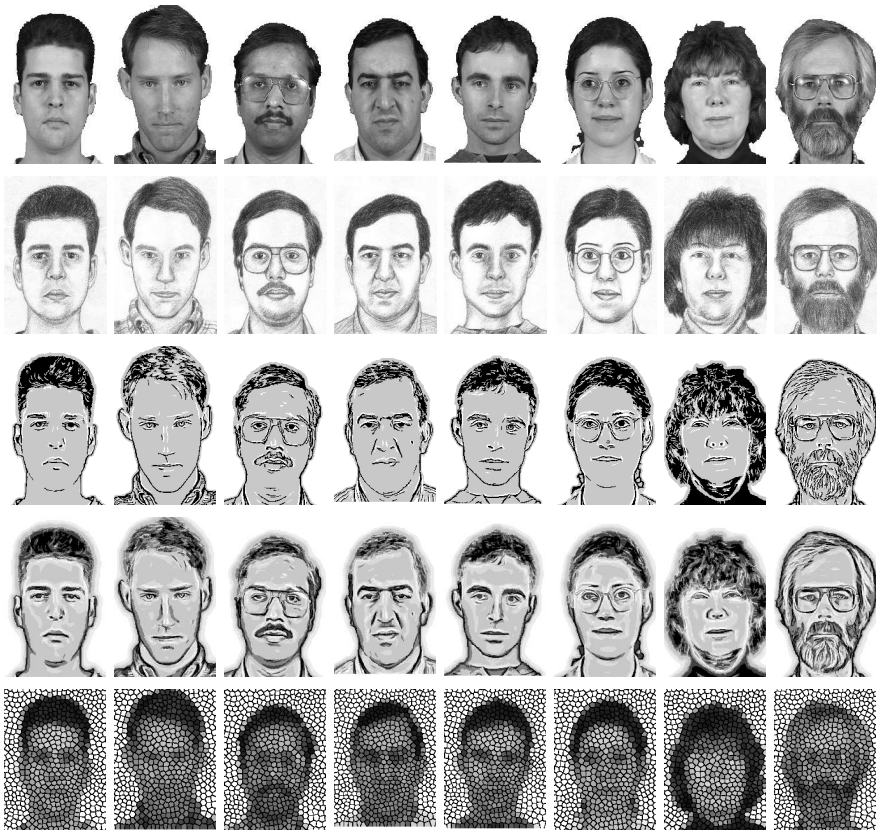

Fig. 29. Examples of stimuli used for the user study. From top to bottom: photos, human artist's drawings, 3-tone rendering, multi-scale pyramid rendering, mosaics.