Richness-Preserving Manga Screening

Yingge Qu Wai-Man Pang Tien-Tsin Wong Pheng-Ann Heng
The Chinese University of Hong Kong

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

Figure 1: Generating manga background from a real photograph. A commercial software (a) adopts the halftone technique and presents a uniform screening result. (b)&(c) are our results with two different parameters. They differentiate the chromaticity in the original color photograph (d) using the variety of screens, while preserving the tone. Both (b)&(c) exhibit 5 types of screens.

Abstract

Due to the tediousness and labor intensive cost, some manga artists have already employed computer-assisted methods for converting color photographs to manga backgrounds. However, existing bitonal image generation methods usually produce unsatisfactory uniform screening results that are not consistent with traditional mangas, in which the artist employs a rich set of screens. In this paper, we propose a novel method for generating bitonal manga backgrounds from color photographs. Our goal is to preserve the visual richness in the original photograph by utilizing not only screen density, but also the variety of screen patterns. To achieve the goal, we select screens for different regions in order to preserve the tone similarity, texture similarity, and chromaticity distinguishability. The multi-dimensional scaling technique is employed in such a color-to-pattern matching for maintaining pattern dissimilarity of the screens. Users can control the mapping by a few parameters and interactively fine-tune the result. Several results are presented to demonstrate the effectiveness and convenience of the proposed method.

Keywords: Non-photorealistic rendering, manga, multidimensional scaling, screening.

1 Introduction

Japanese comic, or manga, is a popular art form and medium of entertainment over the world. It is unique in its elegant use of rich sets of screens, and a tidy and fine drawing style. Screening refers to the process of laying pre-printed patterns (can be regular or irregular) over a region. Manga artists usually lay screens not solely according to shading (tone), but also the texture, material property, or even the chromaticity of the underlying surface being expressed. Such a screening process is tedious and labor intensive. The situation is even worse for complicated backgrounds [Nagatomo 2003].

To save time and cost, some artists have already adopted com-

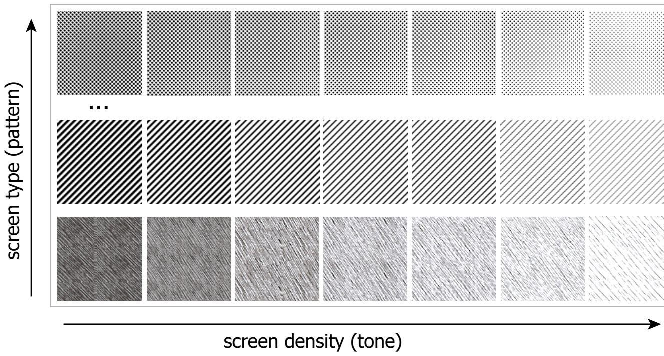


Figure 2: The screen space. Apart from tone variety, another dimension is the texture or pattern.

puter techniques to convert color photographs to manga backgrounds, such as hatching (Figure 3(b)), halftoning (Figure 3(c)) and commercial software (Figure 1(a)). However, all these produce inconsistent style and unsatisfactory results compared to traditional manga, as existing methods solely account for the tone reproduction. Note that the bitonal screens offer not only the 1D tone density, but also the variety of patterns (Figure 2). In other words, the screen space is *2D in nature*. Mapping the input color image to grayscale using sophisticated techniques such as, Color2Gray [Gooch et al. 2005] (Figure 3(b)), does not really tackle the 2D screen selection problem as the mapped grayscale image gives no hint on the selection along the pattern dimension (the vertical axis in Figure 2).

We propose a novel *color-to-pattern* method to select manga screens with the goal of preserving three factors: *chromaticity distinguishability*, *texture similarity*, and *tone similarity*. While the tone similarity is trivially preserved by matching with the density (the horizontal axis in Figure 2) of each pattern, our core contribution is on the preservation of chromaticity distinguishability and texture similarity. The pattern variety (the vertical axis in Figure 2) provides us with a new dimension for distinguishing the chromaticity in the original photographs. For instance, we can represent red and orange regions in the original image with two close patterns, while the red and blue regions can be expressed with two dissimilar patterns.

To do so, we project the patterns (with the same density) from the high-dimensional texture feature space (24D as described in Section 4.2) to 2D (same dimensionality as the chromaticity space) via multi-dimensional scaling (MDS). The 2D projected pattern space and the 2D chromaticity space are then matched by optimizing the texture similarity. As shown in Figures 1(b) & (c), the screening results not only preserve the tone, but also differentiate the chromaticity as in the original photograph. In addition, we developed a system that allows users to interactively control and fine-tune the results. In the rest of the paper, we present several results with rich screens that are comparable to real manga pieces by artists.

2 Related Work

Bitonal Image Generation A straightforward way to produce bitonal images from grayscale/color photographs is halftoning (Figure 3(c)). It exploits the spatial integration of human vision to approximate the intensity over a small local region with only black and white pixels [Knuth 1987; Floyd and Steinberg 1974; Ulichney 1987; Jarvis et al. 1976]. As artifact patterns may appear, some techniques [Velho and Gomes 1991; Naiman and Lam 1996] try to reduce the artifact patterns by adjusting the scanning path over the image. To preserve the edges during halftoning, Knuth [1987] enhances edges in a preprocessing step. Velho and Gomes [1995], and Buchanan and Verevka [1995] approximate edge regions by rearranging the clusters around edges. Recently, Pang et al. [2008] preserve the fine structure by optimizing the structure similarity.

(a) Original (b) Hatching from Color2Gray

(c) Halftoning (d) Our method (6 screen types)

Figure 3: Comparing to the hatching in (b) and the halftone result in (c), our result (d) is more consistent to the style of traditional manga.

Hatching is another technique to produce bitonal images (Figure 3(b)). Winkenbach et al. [1994] generated pen-and-ink illustrations by rendering a geometric scene with prioritized stroke textures. A user-specified "detail indication" can be used to highlight complex textures and minimize clutter. Salisbury et al. [1997] use direction fields to guide the orientation of strokes. Durand et al. [2001] suggest a thresholding model of strokes that can express a rich set of stroke styles (such as pencil, charcoal, engraving) to assist artists in achieving various styles. Grabli et al. [2004] improve the clutter control of lines by estimating the line density. This predicts the visual complexity before rendering and allows simplification.

Images generated by halftoning and hatching, containing more or less uniform patterns, may appear monotonous to the reader; traditional manga instead uses a wide variety of screens to enrich the viewing experience. Our proposed method utilizes the variety of pattern to preserve the chromaticity distinguishability as well as the tone and texture similarities.

Perceptual Preservation Gooch et al. [2005] proposed a technique for converting color to grayscale so that perceptual color contrast can be maintained during the conversion. Their method is based on an iterative optimization method to adjust gray value at each pixel. Mantiuk et al. [2006] suggested a general framework for perceptual contrast processing of many kinds of images including HDR images. These approaches aim at reproducing the contrast in the original gamut to a gamut with a reduced dimensionality.

In contrast, our work of color-to-pattern is an attempt to maintain the perceptual difference among colors during the selection of screens. Note that there is no standard way to quantify the perceptual difference among screens. In fact, the preservation can not even be achieved in a pixel-wise manner as the pattern is not a single pixel, but a neighborhood of pixels.

3 Overview

Traditional manga production starts with drafting outlines and structural lines. Then, detail lines are added and inked to finalize the line drawing. Based on the inked lines, manga artists usually select appropriate pre-print screen sheets to fill regions in order to express shading, tone, texture, or atmosphere. Figure 4 shows the typical workflow of traditional screening. The artist may have to lay screens on many regions by carefully carving out along the region boundary (Step 2). The task is rather tedious and labor-intensive, especially when large amount of irregular regions exist.

Given an input color photograph, our proposed method aims at automating the screen selection process with the goal of richness preservation and style consistency to traditional manga. The system starts by segmenting the photograph into regions. Then, the system can intelligently provide the optimal assignment of screens for different regions. This is done by first projecting the available patterns from high-dimensional texture feature space to the low-dimensional color space, and then optimizing texture similarity. In addition, users are still able to control and/or override the selection of screens via simply tunning a few parameters.

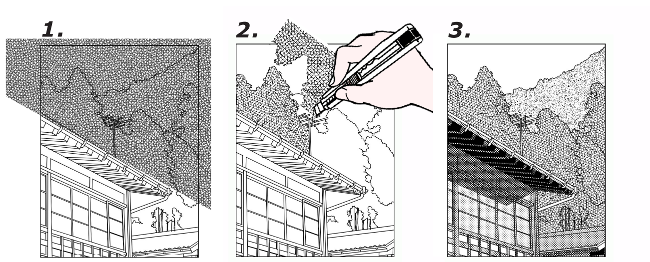


Figure 4: The typical workflow for screening in manga production.

4 Screening

4.1 Segmentation

As screens are laid over regions rather than individual pixels, we first identify regions in the input reference image. We perform the segmentation using mean-shift, which is a non-parametric clustering technique based on the analysis of a complex multi-modal feature space and delineation of arbitrarily shaped clusters [Comaniciu and Meer 2002]. With a kernel measure of the distance between pixels, it robustly produces noise-reduced segments. The mean-shift segmentation result is usually over-segmented, which is obviously too fragmented for direct screening. Thus, we re-group segments according to their color difference and proximity. Near segments with similar colors are grouped first. Note that segments in the same group can still be disjoint. Each segment is then referred as the basic unit in the following screen matching.

Figure 5 shows the segmentation of a region in Figure 1. Note the scene complexity and the large number of segments. The whole image contains 1,563 segments obtained by our segmentation process. With this complexity, semi-automatic segmentation (such as lazy snapping [Li et al. 2004]) plus manual screen assignment could still be very tedious. For instance, the red and disjoint Japanese letters in Figure 5(b) should ideally be laid with the same screen. One cannot select all of them with a single scribble. Instead, they have to be one-by-one selected and assigned to the screen. The selection could be even more tedious if shading exists. In contrast, the proposed screen assignment techniques (described in the following section) can automatically and intelligently assign them with the same screen even they are identified as multiple segments.

(a) Original (b) Segmentation map (c) Our result

Figure 5: For photograph with rich content in (a), thousands of segments can be obtained, as in (b). Semi-automatic segmentation plus manual screen assignment could still be very tedious. The proposed automatic screen assignment can significantly relieve the user intervention and produce reasonable results as in (c).

4.2 Screen Matching

After segmentation, each segment is matched with an appropriate screen from the screen library. Each type of screen (e.g. dot or hatch) in this library has a range of density (tones) as shown in Figure 2. The screens are organized in a 2D matrix with columns of varying density and rows of varying screen type.

The main objective of screen matching is to preserve the rich content in the original reference image. Here, we aim to preserve three classes of contents including *tone* (or luminance), *texture* and *chromaticity distinguishability*. Referring to the two available dimensions in the screen space (Figure 2), tone can be straightforwardly matched with the horizontal screen density axis. However, matching the tone alone does not produce satisfactory results as demonstrated in Figure 6(b). This screening result ignores texture and chromaticity, and more importantly, its style is not consistent with the artworks of professional manga artists. Although the screening style of artists varies, they all enrich the drawing by introducing appropriate pattern varieties.

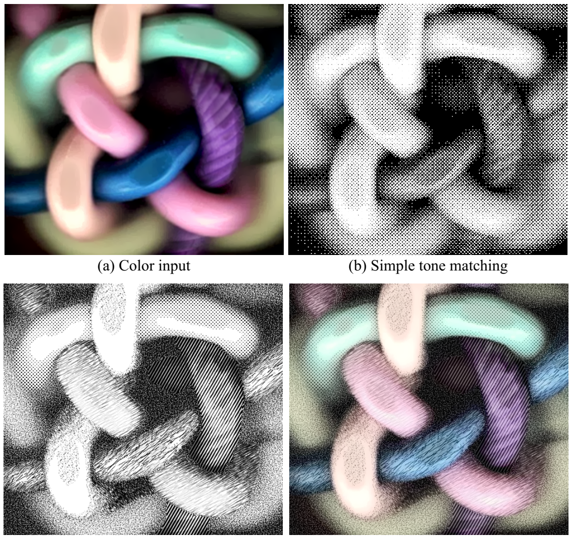
Our screening method consists of the following three steps:

- *Texture-based matching.* Segments containing apparent texture are first matched with screen patterns based on the texture similarity.
- *Color-to-pattern mapping.* Unmatched segments are then mapped to different pattern types with the goal of maintaining color distinguishability, via a color-to-pattern mapping.
- *Tone matching.* Finally, segments assigned to pattern types are screened by matching the tone.

Texture-based matching Segments exhibiting apparent texture characteristics are first assigned a screen pattern based on texture similarity. To quantify the texture characteristics, we compute the texture features on both segments and screens using Gabor wavelets [Manjunath and Ma 1996]. It has been demonstrated as an effective texture identification technique for manga screens [Qu et al. 2006]. Consider an image $I(x,y)$, we compute its Gabor wavelet feature $[\mu_{m,n}, \sigma_{m,n}]_{m=1,\cdots,6,n=1,\cdots,4}$ in a per-pixel manner,

$$\begin{aligned}\mu_{m,n} &= \int\int |W_{m,n}(x,y)|\,dxdy,\\ \sigma_{m,n} &= \sqrt{\int\int \left(|W_{m,n}(x,y)|-\mu_{m,n}\right)^2 dxdy}.\end{aligned} \quad (1)$$

where $W_{m,n}$ are the Gabor wavelets on multiple scales and orientations. In other words, each Gabor wavelet feature is represented as



(a) Color input (b) Simple tone matching

(c) Richness preserved screening (6 screen types) (d) Overlaid with the input color image

Figure 6: Richness-preserved screening.

a 24D vector. For each segment in the reference and each screen in the screen library, we randomly sample one percent of their pixels and compute their Gabor wavelet features. These feature vectors are then averaged to give the representative texture feature for that segment or screen. Note that texture features of screens in the library are precomputed once only as they are fixed.

The matching can be done by measuring the Euclidean distance of the representative texture feature vectors between the segment and the candidate screen. All matching must satisfy (below or equal to) a threshold ε. Segments without such a proper match ($> \varepsilon$) either contain no apparent texture characteristics or no suitable screen available in the library. Hence, they are left to the following color-to-pattern mapping. Figure 8(a) shows an example matching of the textured purple segment with the screen "line."

Color-to-Pattern Mapping Texture based matching addresses the goal of texture preservation. After this matching, many segments remain unassigned due to their textureless nature. So we assign patterns to them by utilizing the chromaticity difference among segments. However, it may not be meaningful to assign, say, a red segment to a hatch screen. To our best knowledge, a direct and apparent relationship between color and pattern does not exist. Hence it is not possible to optimize the distance between color and screen, as such distance metrics may not exist.

In terms of visual richness it would be more beneficial to maintain the perceived distances among different segment colors during the screen type assignment. In other words, we want to maintain the same distance relationship among different color segments after mapping to the screen space. Our screen is in a high-dimensional space (24D) as we mentioned previously while the color is in a low-dimensional space (2D after taking away the luminance). Note that we adopt the CIE L*a*b* color space that consists of one luminance (L*) and two chrominance channels (a*b*). Only the 2D chrominance is considered during the screen type assignment. As mentioned before, the luminance (tone) can be easily matched by selecting the appropriate screen density, once the screen type is selected. This richness preserving screen type assignment is called color-to-pattern mapping.

We employ multi-dimensional scaling (MDS) [Cox and Cox 1994] to maintain such a distance relationship. MDS reduces the dimensionality of the data while maintaining the relative distance among data in the original space (Figure 7). Suppose there are n different

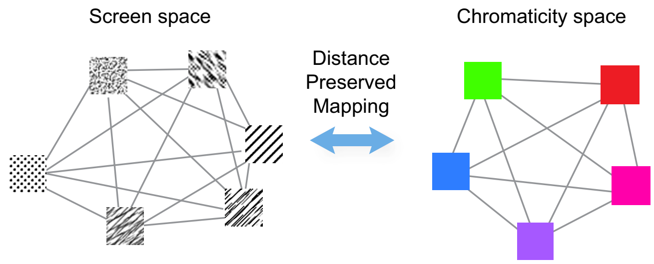


Figure 7: Finding a mapping between the screen space and chromaticity space.

screen types (patterns) in the library. p_i is the m-dimensional feature vector of the i-th screen and d_{ij} is the distance between screens p_i and p_j. Let the 2D point q_i in color space be associated with the screen p_i. Euclidean distance is used to measure the dissimilarity. We further assume that the centroid of all data points lies at the origin, i.e. $\sum_i^n p_i = \mathbf{0}$. MDS formulates the relations between the projected coordinate q_i and d_{ij} in a matrix form as,

$$QQ^T = -\frac{1}{2}[I - \frac{1}{n}[\mathbf{1}]]D^2[I - \frac{1}{n}[\mathbf{1}]] \tag{2}$$

Here, Q is a $m \times n$ matrix with columns being the projected coordinate vectors, D is the dissimilarity matrix which stores the feature distances between all pairs of textures d_{ij} and $[\mathbf{1}]$ is a matrix with all elements being 1. Since D is symmetric, so is the whole right hand side of Equation 2. Hence it can be decomposed using singular value decomposition (SVD), and gives the following equation,

$$\begin{aligned} QQ^T &= V\Lambda V^T \\ Q &= V\Lambda^{1/2} \end{aligned} \tag{3}$$

Here Λ is diagonal matrix with all the eigenvalues. Then, we extract the major components by simply keeping the first two rows and truncating Q in Equation 3 into a $2 \times n$ matrix $\hat{Q}$. Then, the columns of $\hat{Q}$ give all q_i, which are the projected coordinates of the i-th screen in the 2D chromaticity space. Since there is no specific scale relationship between distance in screen space and chromaticity space, all q_i are normalized to $([-1,1] \times [-1,1])$ or based on the range of all available colors on segments.

Once we compute the MDS for all screens, the most straightforward method is to process each segment by first computing its average color in the segment and then matching a screen with the nearest q_i. Note that the total number of distinct segment colors W (in the magnitude of hundreds) is usually much larger than the total number of screen types K (e.g. 1-10) used for a single manga. Manga artists seldom use hundreds of types of screen in one artwork. To maximize the usage of the K available screens, we first perform K-means clustering on the segment colors with K being the number of clusters. Then, the centroid of each cluster is used as a representative color to match with the nearest screen coordinate q_i in color space.

Notice that such mapping is not unique, as the MDS only maintains the *relative* distances. We can still transform the mapped coordinates q_i by rotation, scaling, translation, or flipping. Both scaling and translation can be excluded as the coordinates q_i and segment colors can be normalized and centered (given zero mean). However, there are still two degrees of freedom left. To determine the best transformation, we tentatively perform color-to-pattern mapping on *all* segments including those already assigned with screens in the previous texture-based matching. The rationale is that texture-based matching does not account for colors and, more importantly, we can utilize these pre-assigned segments to *anchor* the transformation. Hence, we optimize the transformation with an objective

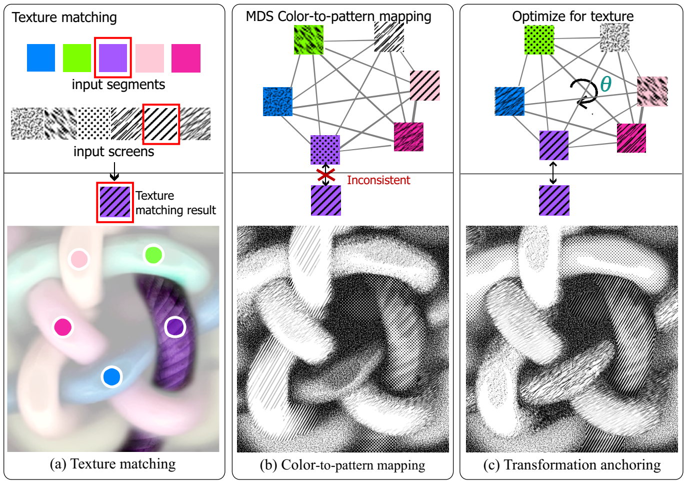


Figure 8: Optimization with texture feature matching and color variations. (a) Texture matching is performed. The purple segment is the only segment matched with a screen with similar texture. (b) Initial screen matching based on color distinguishability. (c) Optimizing θ to keep the color-to-pattern mapping consistent to the previously matched screen.

of maximizing texture similarity of the screens and segments. Figure 8(b) illustrates the initial transformation with the purple segment being assigned to a dotted pattern which obviously does not match well with the texture characteristic of the purple segment (see Figure 8(a)). By maximizing the texture similarity during the adjustment of the transformation, a better screen assignment can be obtained (as Figure 8(c)). The transformation optimization is formulated as follows:

Suppose we have S screen-assigned segments (assigned in the first stage) out of the total $\hat{S}$ segments. τ_s and $C(s)$ are the pre-assigned screen and the average color of segment s respectively. $T(\tau)$ is the texture feature for screen τ. We rotate with angle θ and flip f (takes 1 or -1) to determine a transformation such that,

$$\operatorname{argmin}_{\theta,f} \sum_{s}^{S} (a_s \times |T(Q(\theta,f,C(s))) - T(\tau_s)|) \qquad (4)$$

where $Q(\theta,f,C(s))$ is the screen selected via color-to-pattern mapping given the color $C(s)$, rotation angle θ, and flipping f. a_s is the size of segment in the image and $\sum_s^{\hat{S}} a_s = 1$. If we quantize θ to 360 levels, then the total number of possible transformation is only 720 (360 $\times$ 2). Even an exhaustive search can solve the optimization in seconds. Note that it is possible that the screen patterns selected via the above optimization do not completely coincide with the *multiple* pre-assigned screens from the first stage. In that case, we simply use the pre-assigned screens as we give higher priority to the texture similarity.

Tone Matching After the first two stages, each segment has been assigned to a screen type. The final step is to perform a tone matching by selecting the appropriate screen density. We match the overall grayness of the screen with the average luminance of the segment. Note that the luminance value depends on the color model being adopted during the grayscale conversion. By default, our system adopts the $L^*a^*b^*$ color model and refers L^* as the luminance. However, for color to grayscale conversion, YC_rC_b model is more common and Y is referred as the luminance. Our system allows the user to optionally select the color model for grayscale conversion. Our result in Figure 9 uses YC_rC_b for grayscale conversion, while other examples in this paper use $L^*a^*b^*$.

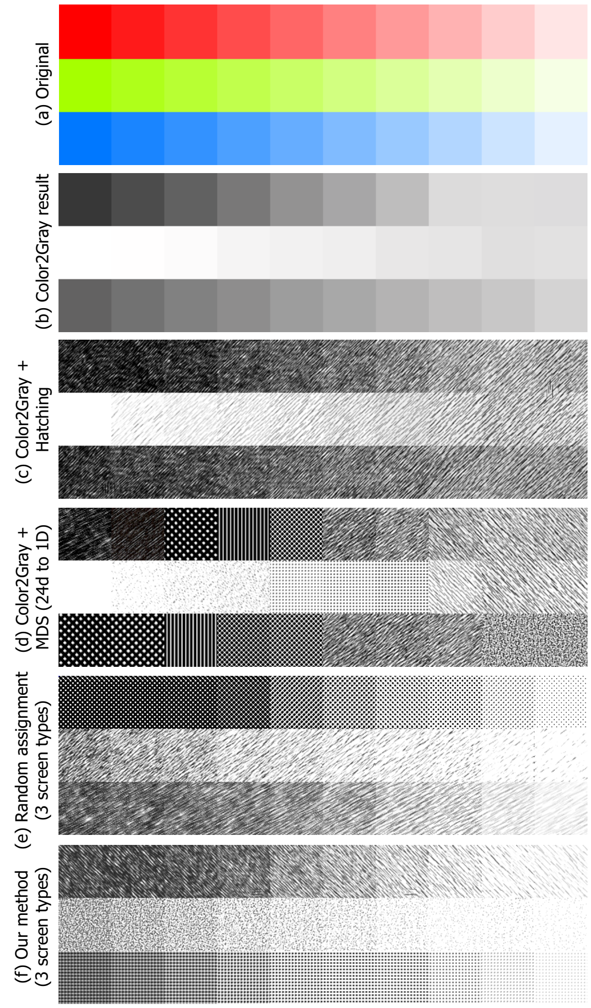
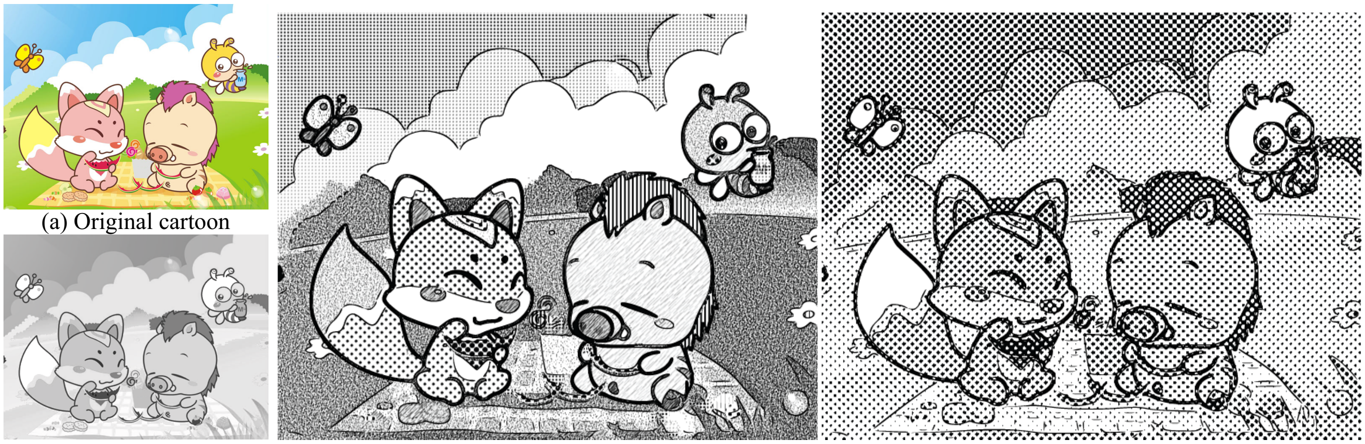


Figure 9: Application of Color2Gray for richness preservation. Without exploiting the two-dimensional nature of the screen space, the extended application of Color2Gray may either produce (c) uniform screening or (d) excessive pattern assignment. The random assignment (e) may lead to unpredictable appearance and suffer from chromaticity indistinguishability. In contrast, our result preserves the chromaticity distinguishability as well as tone consistency.

As the Gabor wavelet texture feature we employed is rotational invariant, we need to geometrically rotate the chosen screen (just like artists physically rotate the pre-print screens) in order to align with the texture characteristics in the segment. Figures 6(c) and (d) show the screens selected by our method without and with the color image laid underneath, respectively. Note how the selected screens faithfully preserve the tone similarity, the texture similarity, and the chromaticity distinguishability.

Discussion One may argue an extended application of state-of-the-art Color2Gray [Gooch et al. 2005] technique is sufficient to obtain the similar result. Figure 9(c) shows an example by screening Figure 9(a) with a hatching pattern, based on the Color2Gray result (Figure 9(b)). This approach only maintains the distinguishability by screen tone, but the screen pattern remains uniform over all regions. The variety of screen pattern is not properly exploited, and preserves no chromaticity distinguishability.

To further verify the applicability of Color2Gray, we project the screens from 24D to 1D (graylevel) and match them with the grayness from Color2Gray result. Figure 9(d) shows the matching re-



(a) Original cartoon

(b) Color2Gray result

(c) Our result (5 screen types)

(d) Halftoning of (b)

Figure 10: Color-to-bitonal result from a color cartoon.

sult. It has more screens adopted compared to the first trial, but multiple screen types are assigned to the region with the same chromaticity. This is because the excessive dimensionality reduction actually collapses not only the chromaticity distinguishability, but also the original grayness into a single dimension. In contrast, our method (Figure 9(f)) utilizes both dimensions of the screen space and preserves the chromaticity distinguishability as well as the tone similarity.

We also compare our method to random screen assignment. In this random assignment, we first cluster all segments into K groups based on their chromaticity in the original photograph, where K is the desired number of screen types given by the user. Then K screens are randomly assigned to the K groups in a one-to-one manner. Such random assignment may result in unpredictable and undesirable screening. As shown in Figure 9(e), a worse assignment may accidentally lead to similar hatching appearance in the second and third rows. The original chromaticity difference is totally ignored and no longer distinguishable after the random screen assignment. Moreover, texture details may not be properly preserved, as demonstrated in Figure 16(c).

5 Results

To verify our method, we test a variety of examples, including photographs, color cartoons, and artworks. Figures 3, 9, 10 and 11 compare our results to those of halftoning or hatching the Color2Gray images. Figures 1, 12-14 show the comparison with the commercial software [CELSYS] results. We leave out the comparison to simple hatching and halftoning, since they are clearly outperformed by the commercial software. In all examples, we obtain mangas with rich screening. The number of screen types used are shown underneath each image. Obviously, Color2Gray and commercial software produce monotonous results that are not manga-like.

Most manga drawings contain clear and tidy line drawings. Unlike the eye-tracking approach to identify important lines [DeCarlo and Santella 2002], our system tries to generate lines by heuristically filtering away detail lines. Starting from the improved edge detection result [Meer and Georgescu 2001], we rank the importance of lines according to their length, curliness, size of associated segment, and visibility value. For the sake of fair comparison, we laid the same set of lines determined as illustrated to both our results and the results by other methods being compared (Figures 10-14, and 16).

Our proposed color-to-pattern matching can be applied to any color-to-bitonal image applications that requires chromaticity distinguishability. For example, color artwork (Figures 3 and 11) and cartoon drawing (Figure 10) can also be converted to bitonal images. In the manga industry, cartoon (anime) may be published, not only in video form, but also in a manga form. Current practices simply print the cartoon key frames in color, probably due to the high and manual color-to-manga cost. With the proposed method, it is possible to publish the color cartoon in bitonal manga form (lower printing cost) in a style consistent with traditional manga.

For an 800×1000 input image, our system can automatically generate the result in about 4 minutes, on a PC with P4 3.2GHz CPU, 2GB memory. This includes the time for segmentation, texture matching, color-to-pattern matching, tone matching, plus the minimal user control.

Limitations Currently, we measure the chromaticity similarity in CIE $L^*a^*b^*$ space and pattern similarity in Gabor wavelet feature space. To our best knowledge, it is still an open problem to understand whether these two kinds of similarities (measured in two different spaces) are linear to our human perceptual distance. Hence, it is also not known whether these two similarity distances are perceptually linear to each other. Our system performs screening on each input photograph separately. If the same object or building appears in two input photographs, our system may not be able to consistently assign with the same screen. A temporal constraint (over multiple inputs) may solve the problem, but this leads to a reduction in the number of assignable patterns.

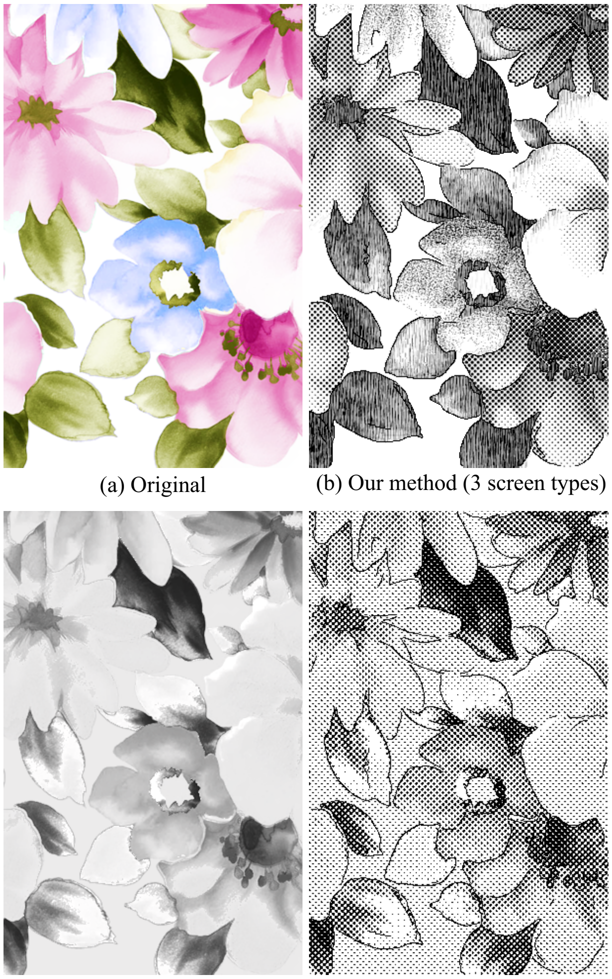
6 Conclusions

In this paper, we propose an automatic yet controllable approach for screening stylish manga backgrounds from photographs. It frees manga artists from the labor-intensive and time-consuming screening process. Mimicking visually rich mangas, the proposed method aims at preserving not only the tone similarity, but also the chromaticity distinguishability and texture similarity. Our results give manga-like style comparable to those manually prepared by manga artists. The proposed color-to-pattern matching technique can be further extended and applied to applications other than manga production.

Acknowledgments

We would like to thank Shree K. Nayar, Daniel Cohen-Or, and Yingqing Xu for their advice on this project. Thanks to all reviewers for their constructive comments and guidance in shaping this paper. Thank to Ming-Te Chi and Wei Xiong for helping screen preparation and helpful discussion. Thanks to Carl Jantzen for the video narration. This project is supported by the Research Grants

Council of the Hong Kong Special Administrative Region, under RGC Earmarked Grants (Project No. CUHK 417107), and the Innovation and Technology Fund of Hong Kong Special Administrative Region (ITS/054/08).



(a) Original (b) Our method (3 screen types)

(c) Color2Gray (d) Halftone from Color2Gray

Figure 11: "Artwork"

References

BUCHANAN, J. W., AND VEREVKA, O. 1995. Edge preservation with space-filling curve half-toning. In *Proceedings of Graphics Interface 95*, 75–82.

CELSYS, INC. Manga Studio. In *http://www.comicstudio.net/*.

COMANICIU, D., AND MEER, P. 2002. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell. 24*, 5, 603–619.

COX, T. F., AND COX, M. A. A. 1994. *Multidimensional Scaling*. Chapman & Hall.

DECARLO, D., AND SANTELLA, A. 2002. Stylization and abstraction of photographs. In *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, ACM Press, New York, NY, USA, vol. 21, 769–776.

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 Techniques*, Springer-Verlag, London, UK, 71–82.

FLOYD, R. W., AND STEINBERG, L. 1974. An adaptive algorithm for spatial grey scale. In *SID International Symposium Digest of Technical Papers*, Society for Information Display, 36–37.

GOOCH, A. A., OLSEN, S. C., TUMBLIN, J., AND GOOCH, B. 2005. Color2gray: salience-preserving color removal. *ACM Trans. Graph. 24*, 3, 634–639.

GRABLI, S., DURAND, F., AND SILLION, F. 2004. Density measure for line-drawing simplification. In *Proceedings of Pacific Graphics*.

JARVIS, J. F., JUDICE, C. N., AND NINKE, W. H. 1976. A survey of techniques for the display of continuous tone pictures on bilevel displays. *Comput Graphics Image Process 5*, 1, 13–40.

KNUTH, D. E. 1987. Digital halftones by dot-diffusion. *ACM Transactions on Graphics 6*, 4, 245–273.

LI, Y., SUN, J., TANG, C.-K., AND SHUM, H.-Y. 2004. Lazy snapping. *ACM Transaction on Graphics 23*, 3, 303–308.

MANJUNATH, B. S., AND MA, W. Y. 1996. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence 18*, 8, 837–842.

MANTIUK, R., MYSZKOWSKI, K., AND SEIDEL, H.-P. 2006. A perceptual framework for contrast processing of high dynamic range images. *ACM Trans. Appl. Percept. 3*, 3, 286–308.

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

NAGATOMO, H. 2003. *Draw Your Own Manga: All the Basics*. Kodansha International Ltd.

NAIMAN, A., AND LAM, D. 1996. Error diffusion: wavefront traversal and contrast considerations. In *Proceedings of Graphics Interface 96*, 78–86.

PANG, W.-M., QU, Y., WONG, T.-T., COHEN-OR, D., AND HENG, P.-A. 2008. Structure-aware halftoning. *ACM Trans. Graph. 27*, 3, 1–8.

QU, Y., WONG, T. T., AND HENG, P. A. 2006. Manga colorization. In *ACM Transactions on Graphics, SIGGRAPH 2006*, vol. 25, 1214–1220.

SALISBURY, M. P., WONG, M. T., HUGHES, J. F., AND SALESIN, D. H. 1997. Orientable textures for image-based pen-and-ink illustration. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 401–406.

ULICHNEY, R. A. 1987. *Digital Halftoning*. MIT Press, Cambridge, MA.

VELHO, L., AND GOMES, J. M. 1991. Digital halftoning with space filling curves. *Computer Graphics 25*, 4 (July), 81–90.

VELHO, L., AND GOMES, J. M. 1995. Stochastic screening dithering with adaptive clustering. *Computer Graphics (Proceedings of SIGGRAPH'95)*, 273–276.

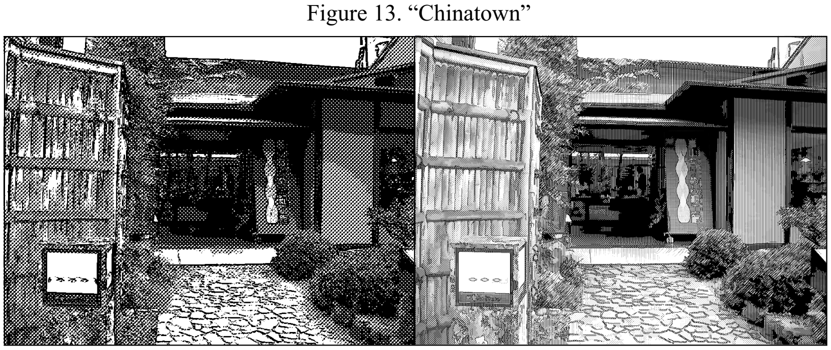
WINKENBACH, G., AND SALESIN, D. H. 1994. Computer-generated pen-and-ink illustration. *Proceedings of SIGGRAPH 1994*, 91–100.

(a) Commercial software (b) Our method (4 screen types) (c) Our method with a different parameter (3 screen types)

Figure 12. “Cafeteria”

(a) Commercial software (b) Our method (5 screen types)

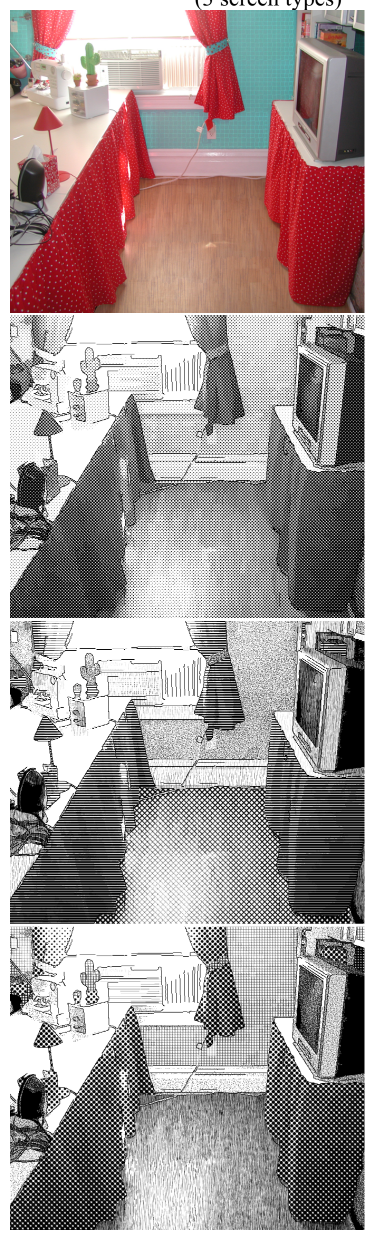
Figure 13. “Chinatown”



(a) Commercial software (b) Our method (4 screen types)

Figure 14. “Tea house”

Figure 15. Original images



(a) Original

(b) Halftoning from Color2Gray

(c) Random assignment

(d) Our method (4 screen types)

Figure 16. “Indoor” with multiple textures on the wall, floor, curtain, and table cloth. Our result (d) properly preserves the original textures.