Abstraction of photographs: a comics style approach

Catherine Sauvaget
L.I.A.S.D.
University Paris 8
2 rue de la liberté, 93200 Saint-Denis
France
cath@ai.univ-paris8.fr

Vincent Boyer
L.I.A.S.D.
University Paris 8
2 rue de la liberté, 93200 Saint-Denis
France
boyer@ai.univ-paris8.fr

Abstract: We present an automatic image abstraction framework that abstracts photographs through a comics style approach. Our model extracts image structures and creates a depth map. Then it automatically adapts line drawing and colorization to stylize the image. Results prove that our model is suitable to create comics images from photographs and is well adapted for efficient visual communication.

Key–Words: Non-photorealistic rendering, visual perception, image abstraction, image processing.

1 Introduction

Image abstraction is the process that reduces or emphasizes some details to focus on a more simple concept.

In computer graphics, image abstraction is one of the most important applications of non-photorealistic rendering in which the rendering depends on the relative importance of various scene elements. Image abstraction systems are designed for efficient visual communication. They try to make images easier or faster to understand. Generally these systems realize a particular stylization to provide abstraction. Unfortunately, the stylization process generates two problems: a problem of loss of depth (objects which have a different depth in the photograph appear on the image with the same one) and they provide only one style.

In graphic arts, comics is generally defined as a graphic medium that consists of one or more images conveying a sequential narrative. Comics creation consists of these successive steps : elaborating a sketch, drawing with pencil, then inking and colorizing. To achieve a comics may be the work of one individual or of a team (plotter, penciller, inker, colorist...). Comics images can be considered as abstraction images. They are expressive, only relevant informations to the purpose are precisely described in the image. Moreover the relevance of details in the image is emphasized by the comics creators style. In addition comics creators have developed techniques to depict the depth in the picture and a lot of different styles exist. So, this kind of stylization is suitable to realize image abstraction in computer graphics.

We propose an automatic image abstraction framework that abstracts photographs by a comics style approach. Our model extracts image structures and creates a depth map. Then it stylizes the image according to techniques used by comics creators.

In the following, we present work related to abstraction and stylization. The main drawback of this work is described: a problem of loss of depth hereafter called the depthless problem. As we propose a comics approach, we study the different comics creator styles. Then we propose our 2D model based on a comics graphic arts analysis to transform a photograph into comics style pictures. Finally we present images produced by our system.

2 Previous works

Image processing techniques are often used to produce artistic style pictures [10], [6], [5]. A lot of previous work has tried to provide image stylization for efficient visual communication. DeCarlo et al. [3] describe a computational approach to abstract photographs and to clarify the meaningful structure in an image. A hierarchical image representation is realized and a rendering with a perceptual model is achieved where regions are filled in with their average color. Regions boundary curves and lines are drawn according to a contrast model.

Figure 1 presents a photograph and the result produced. Remark that the depth seems to be overwritten and it is difficult to precise the distance of the group in the produced image.

Wang et al. [11] add a temporal coherence to produce video stylization but the same problem persists. Winnemöller et al. [12] have proposed an automatic real-time framework that abstracts imagery by mod-

Figure 1: Stylization and Abstraction of a photograph, DeCarlo et al.

eling color opposition contrast and visual salience in terms of luminance.

Figure 2: Abstraction example, Winnemlller et al.

Figure 2 presents a result of this method. As one can see the depthless color problem appears even when the image is blurred.

The main drawback of these works is the appearance of depthless areas: objects which actually have different depths in the photograph appear on the image with the same one. As the image is segmented and the selection of the color used to fill in a region is realized without depth information, the depth disappears. Moreover these works provide images close to a single comics style and it is impossible to produce other ones.

3 Comics analysis

Comics exist in a wide variety of incarnation. There are many comics types and it is impossible to give an accurate definition [8] but different styles can be presented:

- straight stroke: this is the most frequent cartoon style without color shadings or black flats (Edmon Baudoin, Hergé, Tito, Bourgeon, Tabary).

- stroke with black flats: the character is emphasized and the flats permit dramatization of the picture without half tints (Blutch, Brescia).

- realistic cartoon: straight strokes sometimes are blended with many black flats. The purpose is to produce relief and to give a realistic atmosphere with contrasted lighting (a dramatization) (Adrian Tomine, Gillon).

- wash: half-tones for black and white applied by touch with different intensities or by scale. It is used to create a cold atmosphere (Pascal Rabate).

- other styles exist like modeling by stroke (François Schuiten, Bilal, Manara, Moebius. . .) or am gray-tone stroke (Moebius).

When used, colors are very important in comics pictures because they create depth. Artists generally increase color contrast to support the perception of objects in the picture. To produce depth the main principle of the comics creators is to decrease the color saturation according to the depth of the objects in the scene. Moreover, a background color can be used to illustrate objects located at a great distance. As an example, in figure 3, we modify the original picture (hue and saturation of hills) and the depth disappears (are the cypress farther away than the hill on the left?). Remark also on the right, that when the same color is used, the mountain seems closer.

Figure 3: Depth importance: Original, constant saturation, constant tint (from "Quintett" by Cyril Bonin).

Colors can also be used to give an atmosphere (see figure 4). For example, dramatic atmospheres can be produced with browns and dark blues, mysterious atmospheres are generally composed of dark blues and greens, violent atmospheres use oranges and reds, threatening atmospheres are realized mainly with ochers and browns.

4 Our model: CSM

CSM, Comics Stylization Model, abstracts photographs by a comics style approach. Our model is composed of three successive parts (see figure 5). We first extract image structures from the photograph. Then we generate a depth map using them. Finally a stylization process is realized using previous results and a style given by the user.

Figure 4: Examples of atmosphere: (left)"La chutte des mondes" by Alexis Aubenque, illustrator Alain Brion (2003), (right) "Black OP 4" by Desberg and Labiano (2008).
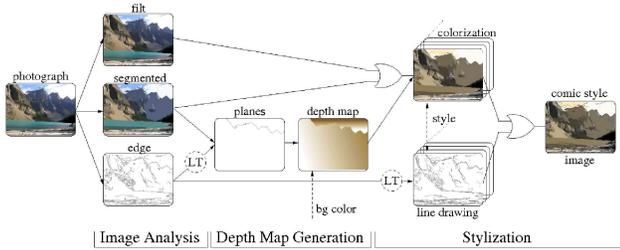

Figure 5: Framework overview: the three successive parts of **CSM** where dash lines represent parameters given by the user.

4.1 Image structure and analysis

As previously described, comics pictures are often characterized by a pen and ink drawing of contours and a colorization by one color (one tint) per region. So, we propose to detect contours with an edge detection method. A mean shift method is used to produce the segmentation and the filt images (terminology proposed by [2]).

4.1.1 Edge detection

Several systems have been proposed for edge detection. We use the method proposed by Meer et al. [9]. A list of contours $\mathcal{L}_C$ composed of n contours is created. Contours have no loops (i.e when a loop or when two possible paths are detected, two contours are created). A length threshold (named LT in figure 5) can be used to remove non-significant contours and then $\mathcal{L}_C$ is updated. This list is only used for the plane detection process.

4.1.2 Segmentation

Segmentation can be considered as a general technique to extract significant image features. We use the mean-shift algorithm [2], a non-parametric procedure for estimating density gradients. The main idea is that the vector of differences between the local mean and the center of a window, named the mean-shift vector, is proportional to the gradient of the probability density at a point. Regions are created in the joint domain by grouping together neighboring pixels which are closer to a convergence point. The regions containing less than M pixels are eliminated and the process is repeated as long as the number of regions given by the user is not reached.

4.1.3 Filt

The mean-shift algorithm [2] can also be used to produce a discontinuity preserving smoothing technique. It is a simplification of the segmentation method where the kernel window moves in the direction of maximum increase in the joint density (i.e. there are no created or eliminated regions). The produced image is more detailed than the segmented one. Features with a high color contrast are preserved independently of the number of pixels in the region, and regions containing pixels with a low color contrast are almost completely smoothed.

4.2 Depth Map Generation

A lot of work has been realized in generating a depth map from an image. Methods based on vanishing-point detection using edge detection are not well adapted to a landscape photograph but more to a city photograph with geometry. [1] estimate the depth information by a color-based segmentation, a specific region detection based on semantic rules and an image classification. The classifier is necessary to obtain a reliable depth map and to tag each part of the picture. An approximated depth map is then generated. Unfortunately, the image classifier is not complete and depth map generation must often be realized interactively. [7] improve this method but some problems remain: when the photograph is not horizontal or when an object like a grid is present, superpixels are created and the plane detection is not correct.

As comics pictures are generally characterized by different planes and a linear perspective, we propose an approach based on a border plane detection.

To detect the border planes, the first pixel of contours in $\mathcal{L}_C$ is selected if the color distance between the two regions is greater than a threshold. This constitutes a list $\mathcal{L}_{FP}$.

Then our algorithm computes a border plane for each selected pixel P of $\mathcal{L}_{FP}$ (the result is stored in a list $\mathcal{L}_{BP}$, note that $\mathcal{L}_{BP}$ is empty at the initialization) as following:

1. Find the contour C_i in $\mathcal{L}_C$ containing P.

2. Starting from P follow C_i and search KP_i (KP_i^1 or KP_i^2 depending on the direction). KP_i can be the first crossing point, the extremity or a point of the border plane list $\mathcal{L}_{BP}$ (in that case, KP_i is not evaluated).

3. Find the next point P':

   - if KP_i is a crossing point, choose the next point P' of the crossing which does not belong to (KP_i^1, KP_i^2) and which has the nearest characteristics to KP_i in the segmented image (see figure 6 (left));

   - if KP_i is an extremity, search its nearest point P'' of C_k ($k \neq i$) in the segmented image which has characteristics close to KP_i (a threshold value based on a color difference is used). If the distance between KP_i and P'' is greater than the distance between KP_i and the border of the image or if no P'' exists then go to the last step (condition 1). Else search the key points KP_k (i.e KP_k^1, KP_k^2) for P'' in C_k. We find P', a point of C_k and the point Q of C_i which minimize their euclidean distance and which belong to KP_k and KP_i respectively. A new contour is created between these two points (see figure 6 (right)).

4. We construct the border-plane list by adding the path between P and P' and we iterate this process (P' becomes P) until condition 1 is verified.

5. Last step: the path is completed by adding a link between KP_i and the border of the image.

The border plane which contains the most pixels in $\mathcal{L}_{FP}$ is selected.
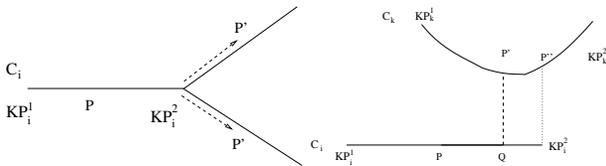

Figure 6: An example of an edge detection of the border plane for a crossing (left) and for extremity (right).

Finally for each plane the user gives a vector and the minimal and maximal values. We fill in each region using a linear interpolation. In our model, each value of the depth map is in [0;1] where 1 refers to the camera viewer and 0 to infinity. Note that a chromatic value can be used to depict an atmosphere.

4.3 Stylization

Stylization is the last step of our workflow. **CSM** realizes both a colorization and a line-drawing image that can be mixed to produce the comic style picture.

4.3.1 Colorization

The user chooses the filt image or the segmented one and the depth map is applied automatically. As comics creators do, we want to decrease the saturation to depict depth and to use the hue of the depth map to give an atmosphere, so we use the HSV model [4]. Let $P(P_h, P_s, P_v)$ be a pixel of the colorization image, $I(I_h, I_s, I_v)$ the pixel of the segmented or filt image and $DM(DM_h, DM_s, DM_v)$ the pixel of the depth map (P, I and DM have the same coordinate). Depending on the style chosen by the user, the saturation of P is computed as:

$$P_s = DM_v \times I_s \tag{1}$$
$$P_s = DM_v \times MaxSat(DepthMap) \tag{2}$$

In (1), the depth map is used to weight the saturation and P_s is in $[0; I_s]$. In (2) we propose to emphasize the depth in the colorization image. In that case, the saturation of I is no more considered to compute the saturation of P. For the other components: $P_h = I_h$ if an achromatic depth map is used, otherwise $P_h = DM_h$ and $P_v = I_v$.

Other effects can be applied:

- Blur (average filters): we use three kernel filters: 3×3, 5×5 and 7×7 according to the values in the depth map and a minimal and maximal distance given by the user.

- Details: as mentionned above, the filt image is more detailed than the segmented image. According to the depth map and depth values given by the user, **CSM** is able to mix these images.

- Binarization: we are also able to produce a binarization (see figure 8), a grayscale or an hatching.

Note that **CSM** (see figure 5) is also able to mix multiple colorization images (an atmosphere, a binarization...) using classical functions (difference, addition, threshold value).

4.3.2 Line Drawing

We use the edge detection image to compute the line drawing image. A length threshold can be applied to select significant contours. Three different styles are available:

- thick contours: each contour is divided into 5 equal parts. We increment the thickness of: the extremal parts by one pixel; the medium parts by two pixels; the center part by three pixels. To avoid the aliasing produced, we apply a 3x3 median kernel filter. We are able to produce thick contours according to the depth: the thickness of the line decreases when the contour is far in the image

- short breaking lines: a length and an orientation are given by the user. We divide each contour and realize a rotation at the center of each breaking line according to these parameters.

- Spline lines: for each contour, we create a list of controls points and a new contour is created using a Catmull-Rom Spline [4]. Straight lines are detected on the contour and control points are created at their junctions. We compute the vectors between two consecutive control points and if two consecutive vectors are similar (i.e the dot product is close to 1), the control point belonging to the two vectors is removed. Moreover when an acute or a reflex angle is detected between two straight lines (i.e the dot product of two consecutive vectors is close to 0), we create a cusp and the control point is duplicated. Then the Catmull-Rom spline is generated because it guaranties that each control point is hit exactly and the tangents of the generated curve are continuous over multiple segments.

4.3.3 Comics style pictures

Comics style pictures are produced by mixing the colorization and the line-drawing images (see figure 5). For example, we are able to produce the styles presented above (see section 3):

- straight strokes are realized with a colorization generally without atmosphere and contours obtained by the edge detection or thick contours.

- strokes with black flats look like the previous ones but we mix the black flats obtained by binarization during the colorization process.

- realistic cartoons are obtained as strokes with black flats but atmosphere is always included.

- wash is realized using the filt image, an atmosphere and one of the line-drawing styles.

5   Results

This section presents images produced with **CSM**. The CPU time needed to produce the comics style pictures depends mainly on the image size (less than one second for a 800×600 image on a Pentium 3.6GHz with 1Go of memory). The top of figure 7 illustrates the depthless problem. Remark that for the leaves of the tree on the right, it is very difficult to see what is in the foreground and in the background when the depth map is not used (segmented). On the contrary, it is very easy to see it while using a depth map. At the second line on the right an atmosphere is added. Catmull-Rom contours are used for the four last images. From third line to the end bottom, from left to right, we present different line-drawing styles: thick line contours, short breaking-line contours with short straight-line colorization, interpolation between the filt image and the segmented one with atmosphere and with blur depending on the depth map, and finally with an other colored atmosphere.

As one can see in figure 8 (left), the depth map produced is quite good. A little problem occurs with the tree on the right. The comic style picture has been produced by mixing the segmented image with the colored depth map and the contours have been created with Catmull-Rom splines. Figure 8 (right) shows different comics styles. The black and white picture illustrates the binarization with hand-like drawing contours (spline lines). The last image is a mix between a colorization and the binarization.

6   Conclusion

We have presented an automatic image abstraction framework that abstracts photographs by a comics style approach. It is based on an image analysis and structure, that allows to construct a depth map used to perform the stylization process. Compared to previous works, **CSM** provides images without the problem of loss of depth and many different styles are available. Future work will be done to increase the number of comics styles available. We will also try to apply this model to a movie in which temporal coherence remains a problem.

References:

[1] S. Battiato, S. Curti, M. La Cascia, M. Tortora, and E. Scordato. Depth map generation by image classification. In *Proceedings of the SPIE, Volume 5302, pp. 95-104 (2004).*, pages 95–104, 2004.
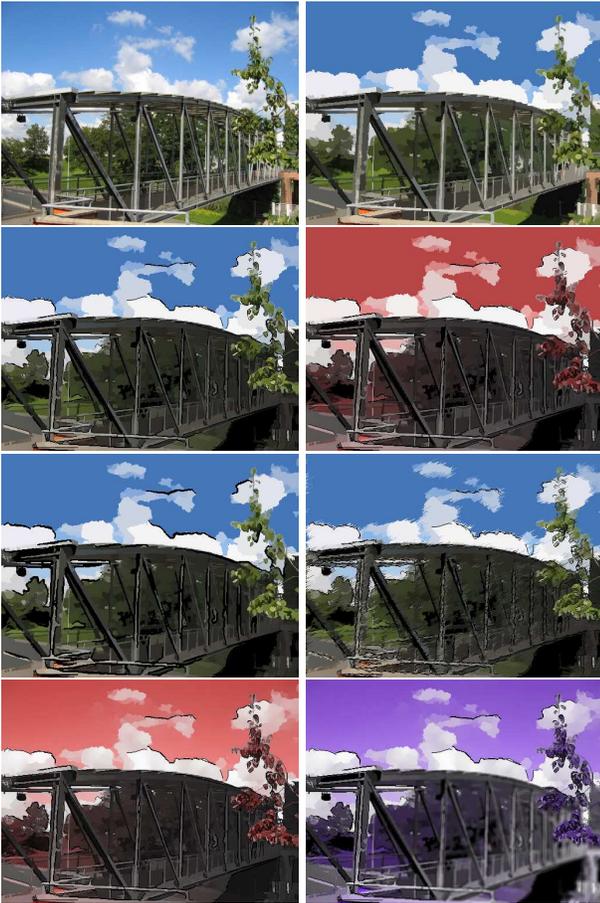
Figure 7: Results on a bridge. Top: original; segmented. Line 2: with depth map; with atmosphere in the depth map. Line 3 to the end: line drawing and post-treatment colorization examples.

Figure 8: (left) Results on a landscape: original, depth map, filt with atmosphere. (right) Results on a face (movie "Slevin"): original; B & W; strokes with black flats.

[2] Dorin Comaniciu and Peter Meer. Robust analysis of feature spaces: color image segmentation. In *CVPR*, pages 750–757, 1997.

[3] Douglas DeCarlo and Anthony Santella. Stylization and abstraction of photographs. In *SIGGRAPH*, pages 769–776, 2002.

[4] J.D. Foley, A. Van Dam, S. Feiner, and J. Hughes. *Computer Graphics, Principes and Practice*. Addison Wesley, 1990.

[5] Aaron Hertzmann. Paint by relaxation. In *Computer Graphics International*, pages 47–54, 2001.

[6] Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless, and David Salesin. Image analogies. In *SIGGRAPH*, pages 327–340, 2001.

[7] D. Hoiem, A. A. Efros, and M. Hebert. Automatic photo pop-up. *ACM Trans. Graph.*, 24(3):577–584, 2005.

[8] S. McCloud. *Making Comics: Storytelling Secrets of Comics, Manga and Graphic Novels*. Harper Paperbacks, 2006.

[9] P. Meer and B. Georgescu. Edge detection with embedded confidence. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(12):1351–1365, 2001.

[10] S. M. F. Treavett and Min Chen. Statistical techniques for the automatic generation of non-photorealistic images. In *Proc. 15th Eurographics UK Conference*, March 1997.

[11] Jue Wang, Yingqing Xu, Heung-Yeung Shum, and Michael F. Cohen. Video tooning. In *SIGGRAPH*, pages 574–583, 2004.

[12] Holger Winnemöller, Sven C. Olsen, and Bruce Gooch. Real-time video abstraction. *ACM Trans. Graph.*, 25(3):1221–1226, 2006.