Vis Comput (2012) 28:877–887
DOI 10.1007/s00371-012-0702-3

ORIGINAL ARTICLE

StoryWizard: a framework for fast stylized story illustration

Jiawan Zhang · Yukun Hao · Liang Li · Di Sun · Li Yuan

Published online: 21 April 2012
© Springer-Verlag 2012

Abstract A fast stylized framework used for creating illustration of a given story is represented. The framework automatically searches proper online images according to the keywords of story, provides users some tools to make the search result precise, and helps users create a picture in every scene without considering the consistency of each character. With the friendly user interface, it provides user abundant interactions, which helps users express their ideas flexibly. Experimental results indicate that this framework allows users, without any art background, to produce personalized picture series with specified story. The fast process, effective interaction and generated delicate pictures can make a story more impressive.

Keywords Storywizard · Story illustration · Image filtering

1 Introduction

Storytelling plays a central role in human experience. People they have used stories to convey information. It is common for readers to depict the written scenes and create images in their minds. Sometimes, in order to spark readers imagination, story illustrations are used to better engage the reader into the story [32]. In recent years, computer software has been widely used by authors to create visual illustrations. The World Wide Web facilitates the sharing of digital photos and has promoted the process of digital storytelling. And the story is told with a set of pictures and a few added comments; the human mind fills in the gaps [10].

J. Zhang · Y. Hao · L. Li (✉) · D. Sun · L. Yuan
Tianjin University, Tianjin, China
e-mail: allen.liliang@gmail.com

Some of the previous works try to search for a suitable picture to illustrate the story [6, 20, 28], others provide a tool to compose a new picture for the specified text [23, 24], but users might not find a proper one in some cases or have no patience on the long composing process. StoryWizard integrates these two methods; we can firstly search suitable pictures based on the online picture collections, and then compose them and create a new one for specific scene.

To carry on story illustration, a main task is to obtain a semantically translation from stories (natural language) to image sequences, which is known as Natural Language Processing (NLP) [1]. The research on NLP has been well studied, so we just need to seek an effective solution meeting our needs. In general, we use an efficient and robust parser to extract the syntactical structures from the input sentences. Based on the structural information, we generate searching queries to obtain images as specialized as possible for each part of the sentence. Typically, for each query, a sequence of images are returned.

The prevalence of the Internet and digital cameras has helped create all kinds of personal and public photo collections on sites such as Facebook and Flickr, from which massive images are now freely available online. Currently, several researchers have also demonstrated by using online photo collections in image editing [4, 19, 31], their approaches are more effective and accurate than traditional ways. This inspires us to create visual story representations from online images. Instead of using a set of fixed images limited to a user-provided database, we aim at using images as broadly as possible, and thus can produce a wide range of visual story representations.

Nevertheless, most of these searching results are unsatisfying, so another key question is how to exclude unsuitable images and express the user's intention. The probably most effective approach is to sketch some flexible and intu-

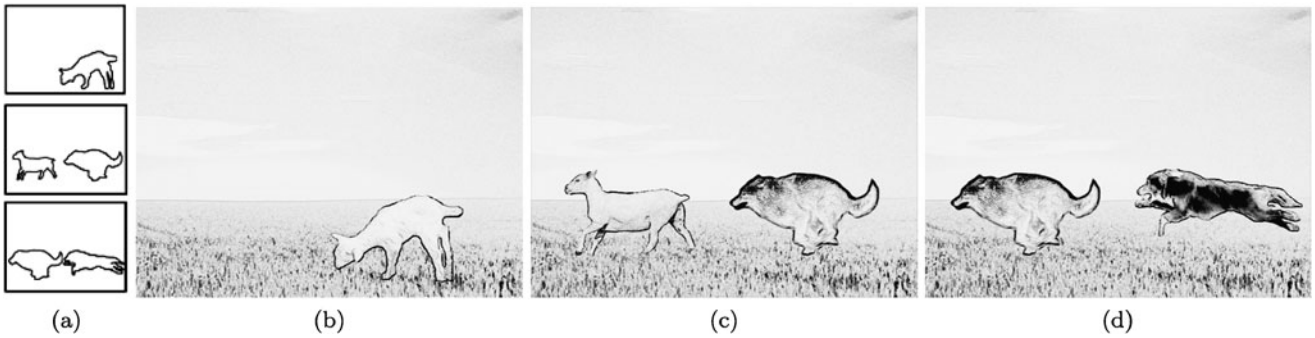


Fig. 1 The visual storytelling results on the "Wolf and sheep" story. (**a**) The user-drawn sketches for the story. (**b**) A sheep eats grass on grassland. (**c**) Suddenly, a wolf comes and chases the sheep. (**d**) Moments later a dog drives the wolf away

itive strokes to express whatever in the user's mind. To our knowledge, most of the sketch based applications focus on image or video retrieval [7, 12]. For finding a similar image, it is difficult or nearly impossible to achieve good search results without a known query image. Search by sketching, however, can naturally address this problem.

In addition, the vividly searching results bring a problem: the consistency of composed picture series. The inconsistencies include light, color, shape, and texture. StoryWizard solve this problem in two ways: (1) It provides a series of image filtering mechanisms. The search results have more similarity in light, color, shape, or texture. (2) It provides a stylized rendering processing step. Through this step, the composed picture series have a better consistency. The stylized rendering processing include hatching rendering [16], painterly rendering [33], water color rendering [8], and pencil rendering [18].

StoryWizard is a story illustration framework driven by a short story, through which a set of images can be semi-automatically obtained from online photo collections, and user-generated sketches that enables users to further express whatever they want. And the scene consistency of each picture is well maintained. Figure 1 demonstrates a visual story representation generated by our system.

In general, the main contribution of this work can be summarized as follows:

- We present a framework for quick visualizing stories in the form of computer graphics. Guided by the framework, users could create visual story representations from online photo collections with simple hand drawing sketches and some interactions.
- We design a user-friendly interface that supports multiple interactive actions. Users can flexibly express their design even without artistic ability, which makes writing story fun for children as well as for screenplay writers.
- We propose an optimized manner to solve the scene consistency problems. After the non-photorealistic rendering, users could get a series of black-white pictures which maintain color, light, texture, and semantic consistency.

2 Related work

Although a number of papers have dealt with issues involved in story illustration [15, 23], most existing techniques rely on text understanding module to interpret the input sentences and construct a set of visual representations based on a user-provided database. For instance, Liu and Leung [23] develop a system called ScriptViz, allowing users to visualize screenplays automatically via realistic, animated 3D graphics. By performing natural language processing, it constructs plans of action and renders scenes, based on a few well-formed, unambiguous English sentences. More recently, Schwarz et al. [28] present a method to semi-automatically create a storyboard from online images, however, the query results cannot insure the visual consistency without considering the spatial relations in a scene. Understanding these methods, we quickly realize how very important the image variety is to the story creation and also seek to build on our story visualization work based on online photo collections.

Recently, several researchers have also demonstrated the use of online photo collections for image creation. Johnson et al. [19] allow users to quickly create a semantically valid image using only text or example pixel patches at different locations. Lalonde et al. [21] use a vast object library that includes all the required object properties to insert new objects into existing images. Chen et al. [4] develop an image montage system, i.e. Sketch2Photo, to synthesize a realistic image by drawing some sketches annotated with text labels. Assisted by the tag, it is able to find the visual object corresponding to the user-drawn sketch from the Internet. Instead of trying to synthesize a realistic image from existing online photos, Wang et al. [31] propose a bilateral interactive image search system called MindFinder, allowing users to sketch and tag query images at the object level. By dragging and dropping objects from search results, MindFinder provides an effective way to return the most similar images to the picture in users' mind. Generally speaking, more semantic information are employed, better results can be achieved.

We extend this line of work with an important distinction. That is, the works discussed so far do not refer to how the online images are semantically translated from natural language, and we believe a system should be able to obtain appropriate image sequences with an effective information extracting and filtering scheme. This belongs to the field of natural language processing [1]. To our knowledge, an efficient and robust parser is essential to any Natural Language Processing (NLP) system. Among all these works, we highlight [29] that utilizes a bottom-up probabilistic chart to extract the syntactical structures from the input sentences. Based on the structural information, the parser can resolve the subjects and the objects, and interpret their meanings.

Our work is also related to content-based image search. Some comprehensive surveys of this field are given by [9, 30]. Since tremendous works have been proposed on searching images via tags and colors, we mainly focus on the sketch-based search which is also applied in our work. In work based on this, Eitz et al. [12] has proposed a novel PhotoSketch system, which takes sketches drawn on a graphic tablet as queries to search in a collection of 1.5 million images downloaded from Flickr. It allows the composition of new images with the found result images and the help of semi-automated segmentation. In our work, the need is much easier, because we do not seek to achieve accurate understanding of each image.

3 Our method

In this paper, we introduce a novel pipeline to efficiently search and create visual story representations, allowing their flexible manipulation. Figure 2 shows an overview of our system. Guided by a natural language processing method, our system decomposes a given input story into suitable scene units (each scene unit can finally correspond to a result picture) and constructs meaningful search terms (keywords). Using these search terms, we search images from online photo collections. The user can optionally draw a sketch to express their intentions.

Once the sketch is finalized, the system starts to compose a stylized picture by searching the appropriate images that match the corresponding search keyword and the provided sketch. We adopt a specific filtering strategy to exclude undesirable images among previous candidate results. For each selected image, we also present a novel algorithm to generate an artistic rendering effect, in which both the structural and salient detail feature can be well retained.

3.1 User interface

As a good user interface is critical for achieving the goal of story creation and visualization, we will briefly describe the design of our user interface at the beginning.

The screen-shot of our UI is depicted in Fig. 3. When the user inputs a story in the text panel, the system can decompose it into suitable scene units automatically. If users are dissatisfied with this default result, they can re-select and repartition the text to generate their own appropriate scene units. Once the user clicks a certain scene unit, the system highlights the keywords for this scene immediately. The left-hand side panel displays the images obtained from the Flickr as the user clicks a certain highlighted keyword. The central panel consists of two parts: the sketch field for drawing scene objects, and the result field for generating a scene. For each keyword, the user can optionally draw a sketch on the sketch field to specify the location, scale and contour of scene object. The user can easily switch these two parts during the manipulation process. Of course, the user can also drag the scene object to change its position or adjust its scale if desired. Furthermore, the system allows the user to interactively refine the composition results. In our system, each scene unit finally corresponds to a result picture which is kept in the top panel.

3.2 Story parsing

Although the story's form is literary; however, the words therein describe pictures and actions. As the story unfolds to reveal the events, we think about translating the script into a storyboard which consists of series of key shots, and we call this process as story parsing. To carry on story parsing, we take a story as input and automatically decompose it into several sentences based on the punctuation by default. Each sentence means one shot, corresponding to a scene picture of the final result. The users can also re-select and re-partition the story to generate their own appropriate scene units.

Since online photo collections provide a great amount of images, users cannot directly find an exact image just by a sentence. So far, the most frequent category of tags in public search engines is words, thereby the main task is to obtain a semantically translation from natural language to image sequences.

Specifically, we use the Apple Pie parser posed in [29] to extract the syntactical structures from the input sentences. Based on the structural information, the parser can resolve nouns, verbs, and adjectives. By default, we use nouns for image search. However, searching by a single noun omits the information of the motion, so it is often helpful to include appropriate verbs or adjectives, for example, 'horse run' and 'cat sleeping'. Therefore, we can combine nouns with verbs or adjectives to retrieve a more specialized collection of images if desired.

3.3 Image filtering

When users click a certain search keyword, the system automatically starts to search images from Flickr. However, web

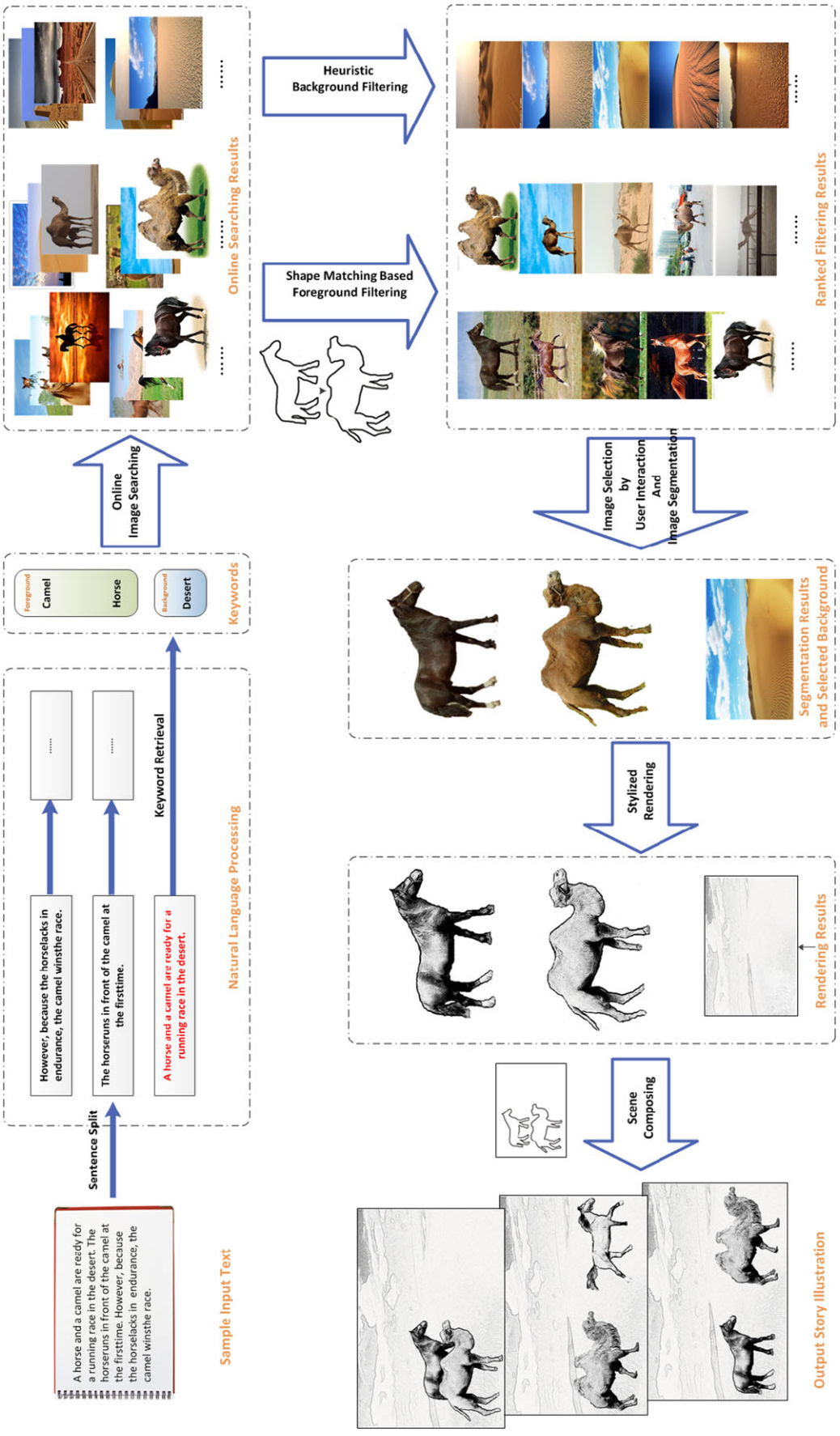


Fig. 2 An overview of our system

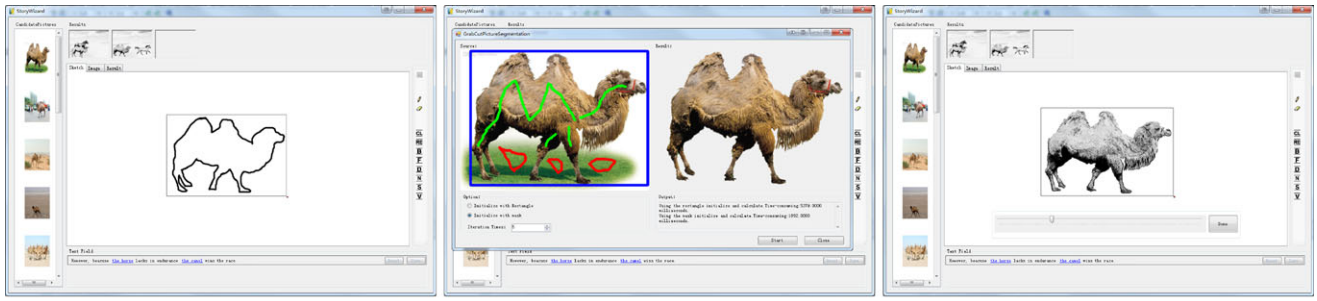


Fig. 3 User interface for our system. (**a**) Sketch pad for drawing freehand sketch to filter the candidate images. (**b**) Separate window for GrabCut. (**c**) Parameterized NPR Processing interface

search often generates inappropriate image results so filtering images discovered from the Internet has been a challenging problem. Here, we adopt a specific strategy to filter both background and foreground images.

Background image filtering In order to highlight the main character in a scene, our system selects background images based on two criterions. Firstly, the background content should be consistent with the search keyword. Secondly, the background image should be generally simple, providing enough space for the foreground objects.

Most often, background images with the same content have similar spatial layout. If we infer a representation of query in a structural feature space, images with similar composition typically cluster together. As one choice of representation that has been used successfully for image recognition and categorization, the gist scene descriptor [26] provides a statistical summarization for the spatial layout of an image. Then we employ the mean shift algorithm [14] to find the largest cluster for the images with consistent composition, based on the gist descriptor. The normalized Mahalanobis distance to the biggest cluster is computed for sorting, and returning the most relevant images (we set 100 in our experiment) for the next stage.

For the rest of the images, we apply a standard segmentation method [13] to perform further filtering, retaining images with large uniform regions. By segmenting each image and calculating the normalized number of segments, the fewer count images corresponding to more simple backgrounds can be retained. This segment count is linearly combined with the normalized Mahalanobis distance, using a weight of 0.2 in our experiments. The system then resorts the remaining images and returns the top 50 results. The effect of this filtering is demonstrated in Fig. 4.

Foreground image filtering For foreground filtering, we mainly consider shape similarity. Firstly, we use a global contrast based saliency extraction algorithm [5] to automatically extract high-saliency segmentation mask for each candidate image, avoiding manually inputting a rectangular region to initialize a segmentation process.

Secondly, we utilize a classical shape matching method to select images corresponding to the provided sketch. Because the segmentation often generates closed regions, we convert the user drawn sketch to closed regions by the morphological close operator. We employ the shape context descriptor proposed in [2] to measure the similarity cost between the user-drawn contour and the salient object contour (extracted by our high-saliency segmentation). This similarity cost is normalized to a value between [0,1], where the most/least similar image has a cost of 0/1. The segmented foreground can be ranked by this cost, and those ranked below 100 (out of 2000) are discarded. Figure 4 demonstrates the effect of this filtering with several foreground objects.

Noting that the saliency-based segmentation is not very accurate comparing with the ground truth of image. However, it avoids manual image segmentation which is time-consuming and impractical, and then guarantees the automatic operation during our shape filtering process. Furthermore, we allow the refinement of automatic segmentation. Once the user chooses a certain foreground image, he can interactively refine the segmentation results with method mentioned in [27].

3.4 Scene generation

Once we have a set of candidate images for each scene object and the background, we can compose these independent images together to generate a complete scene, in which the user-provided story can be better visualized. In order to exhibit a coherent story, we stylize all the images into an artistic effect. Our image stylization algorithm is motivated by Bhat et al. [3] and Wang et al. [18]. We unify these two methods to enhance the salient structure, and preserve the basic details as well. Figure 5 illustrates the results from our basic approach.

Now we can focus on the image composition process. To our knowledge, automatic image segmentation cannot be

Fig. 4 Filtering results. (**a**) Filtering of background and foreground images based on the 'tiger and cat' story. *Top* to *bottom*: discovered background images for the keywords 'forest' and 'tree'; discovered foreground images for the keywords 'tiger' with three different sketches, and 'cat' with two different sketches. The selected images are marked by a *red box*. (**b**) Filtering of the 'camel and horse' story with the same layout as (**a**). The keywords are 'desert', 'camel' and 'horse', respectively. (**c**) The comparative results without (the *top two rows*) and with (the *last two rows*) scene style consistency

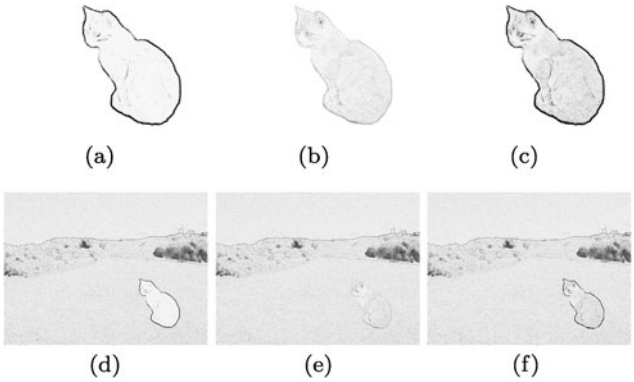


Fig. 5 The comparative results on the 'cat' image. (**a**), (**d**) Result of Bhat's method. (**b**), (**e**) Result of Wang's method. (**c**), (**f**) Our method

guaranteed to work reliably for all situations in spite of a lot recent advances in this field [22]. On the other hand, blending techniques [17] can achieve impressive results without an accurate extraction of the object. More recently, the idea of combining the advantages of blending and alpha-matting has been introduced to hide matting artifacts as mentioned in [4]. Rather than solve this problem in general, we seek an effective solution tailored for our need. We use a content-aware image copy and paste technique [11], which combines ideas from both matting and gradient-based methods. The composition result can be seen in Fig. 6.

3.5 Scene consistency optimization

Although we can separately generate each single scene image that matches the corresponding scene in a specific story, there are still visual consistency problems amongst these scene images, which cannot be figured out well by simply connecting neighboring scene shots together. To solve this problem, we adopt two solutions, one is handling efficiently with recurring queries, and the other is preserving the style consistency.

Scene object recurrence Depending on the visual need associated with these consistency problems, probably the most effective strategy is to apply the selected object for similar sketches appeared in different scene spots. Our system therefore reuses the selection results for scene shots with the same keyword and similar sketch query. When the user finalizes a new sketch having the same search keyword used before, the system utilizes the shape matching algorithm proposed in [2] to measure the shape consistency between the new and the existing sketch with the same keyword. If the shape similarity cost is less than a threshold (we set this value to 0.1 in our experiments), the system will request the user whether or not to reuse the selection object or not. If the user chooses to search new images, all others in this category are updated accordingly.

Style consistency It is widely accepted that the object with different motions should appear in a similar style, based on the same keyword query. Though stylized image effect can eliminate inconsistencies to some extent, the objects with the same keyword in different scenes still vary largely in color and texture, which seems unreasonable in visual aspects. To this end, we consider adding both color and texture features to the existing shape feature.

Assume that A and B are the different scene objects that mapped the same object in a story, the goal of our task is to find a set of images, the color, and texture of which are similar to A, and the shape is similar to B. By ranking these candidate results, we obtain the target image as T. We then formulate our similarity ranking problem into the following optimization:

$$\min D_c(\mathrm{A},\mathrm{T}) \cdot D_t(\mathrm{A},\mathrm{T}) + \lambda D_s(\mathrm{B},\mathrm{T}) \tag{1}$$

where $D_c(\mathrm{A},\mathrm{T})$ denotes the color consistency distance between A and T, $D_t(\mathrm{A},\mathrm{T})$ denotes the texture consistency distance between A and T, and $D_s(\mathrm{B},\mathrm{T})$ denotes the shape similarity cost between B and T. λ is a constant. The default value is set to 0.4 in our experiments. The user may adjust this value to emphasize the shape or alternatively the appearance.

Shape consistency In order to measure the shape similarity between B and T, we calculate the shape distance mentioned in [2]. This distance is a weighted sum of three terms defined as

$$D_s = \alpha D_{ac} + D_{sc} + \beta D_{be} \tag{2}$$

where D_{sc} is the shape context distance defined as the symmetric sum of shape context matching costs over best matching points, D_{ac} means the appearance cost defined as the sum of squared brightness differences in Gaussian windows around corresponding image points, and D_{be} is assigned to be the bending energy. Moreover, α and β are weighting parameter. In our experiment, we set them to 1.6 and 0.3, respectively.

Texture consistency To take advantage of the texture information, we extract texture feature using Gabor wavelet method [25]. Given an image $I(x, y)$, its Gabor wavelet transform is given as

$$W_{p,q}(u, v) = \int_{\Omega} I(x, y) f^*_{p,q}(u - x, v - y)\, dx\, dy \tag{3}$$

where $f_{p,q}$ is the self-similar filter dictionary defined in [25]; superscript * denotes the complex conjugate; and subscripts p and q index the scale and orientation, respectively. We then compute the mean $\mu_{p,q}$ and the standard

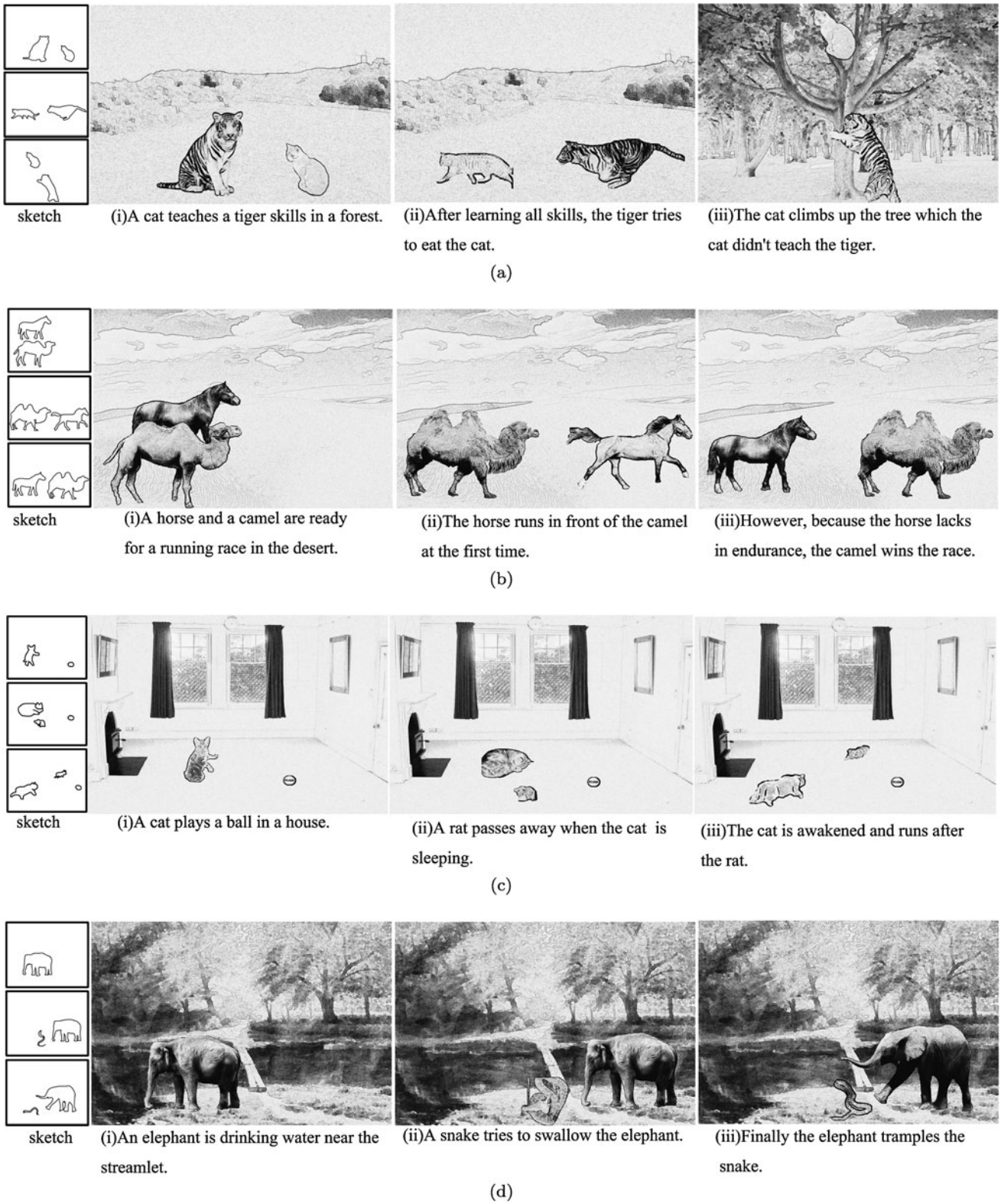


Fig. 6 The visual storytelling results. The *first column* shows the user-drawn sketches according to the story. The rest columns show the story representations generated by our system

deviation $\sigma_{p,q}$ of the magnitude of the transform coefficients:

$$\mu_{p,q} = \iint \left| W_{p,q}(u,v) \right| dx\, dy \tag{4}$$

$$\sigma_{p,q} = \sqrt{\iint \left(\left| W_{p,q}(u,v) \right| - \mu_{p,q} \right)^2 dx\, dy} \tag{5}$$

The feature vector is composed of $\mu_{p,q}$ and $\sigma_{p,q}$ of multiple scales and orientations. In all of our experiments, we use four scales $P = 4$ and six orientations $Q = 6$ inside a $w * w$ window, where w is normally 16. The feature vector formed is

$$T = [\mu_{0,0} \quad \sigma_{0,0} \quad \mu_{0,1} \quad \ldots \quad \mu_{3,5} \quad \sigma_{3,5}] \tag{6}$$

We then compute the Mahalanobis distance between pairs of feature vectors. This normalized distance is regarded as the final texture consistency.

Color consistency To guarantee the color consistency between A and B, we employ a color feature descriptor to exclude severe color miss-matching images, based on A's color feature. For each candidate image, we transform it from RGB color space to HSV color space and use color histograms to obtain image features. The color consistency is computed as the normalized Mahalanobis distances, which is linearly mapped so that the largest and smallest distances are 1 and 0 respectively.

Finally, each segmented foreground object is assigned a total consistency score consisting of shape, color, and texture. The effect of this filtering is demonstrated in Fig. 4(c).

4 Results and discussion

To prove this system's utility, we have tested our system with several examples. Using a short story and a simple sketch, our system can compose a set of story representations that visualize users' idea. To achieve this goal, our system automatically downloads 3,000 images from Flickr and Google for each selected keyword extracted from a user-provided story. The system performs foreground image segmentation while downloading images. To reduce the image sets to a controllable size, 100 foreground and 50 background candidate images are selected, respectively. Figure 1 shows a sample visual story representation generated by our system. The left-most column shows the user-drawn sketches according to the story, while the rest columns show the story representations matching the corresponding scene. To generate this result, we applied all stages of our system. As discussed in Sect. 3.5, if the scene object has a potentially similar shape used before, such as the 'wolf', it is better to perform scene object recurrence. Here, we directly reuse the 'wolf' image in scene 2 into scene 3. As observed, the output result looks natural as desired.

More results are shown in Fig. 6. For a uniform layout, all the showing results are three-frame stories, but we do not have any constraints on the story type or size. Note that the images are chosen specifically in order to evaluate the ability to deal with such challenging problems as scene object recurrence and style consistency. All these scene images have content consistency with the story and a similar contour to the sketch.

However, there are still some limitations which might prevent the users from obtaining satisfactory results through our system. As the saying goes 'there are no two leaves exactly same in the world', we cannot guarantee to find the two exactly same objects from the online photo collections, even if we consider the scene consistency factors in an optimized manner. On the other hand, we do not take perspective problem into account for image filtering and generation. Hence, it may cause some significant artifacts. Furthermore, if the search keyword is a name such as 'Tom' and 'Jerry', our system may fail in this situation and generate incorrect results.

We also evaluate StoryWizard by user study. 25 participants are invited to test StoryWizard. After all of them used StoryWizard, they are asked to finish a questionnaire. The questionnaires include six aspects of StoryWizard: *Is it Useful? Is it Interesting? Convenient to use? Are you Satisfied with processing speed? Are you satisfied with composed results? Are you satisfied with GUI?* Figure 7 shows the number of people who said "Yes" to the corresponding questions. From the user study, we conclude that most people think StoryWizard is useful and are satisfied with the GUI of StoryWizard.

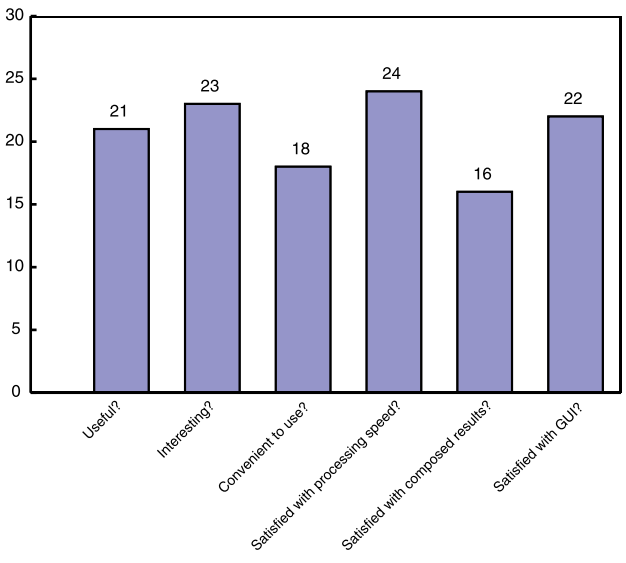


Fig. 7 The user study report

5 Conclusion and future work

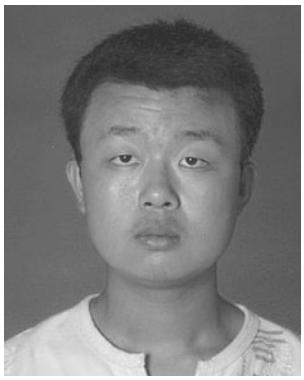
We have proposed a complete method for online story illustration. After parsing the story text, the system automatically generates search and downloads images from the public community site. To further express user's design need, the system allows to draw a simple sketch that corresponds to the specific search object. Most often, a set of representative images are found automatically, based on an effective filtering scheme. After a few user interactions, a reasonable story representation is produced. Owing to a friendly user interface, the system provides users a smart platform for visual feedback, it is indeed a powerful tool in content illustration.

To improve upon these results, future work will have to consider other operations, perhaps perspective adjustments. One natural extension to the presented system is to animate the retrieved images to better visualize the action in a story. Another interesting avenue of future work would be to extend our approach from the still story visualization to the area of video.

References

1. Allen, J.: Natural Language Understanding, 2nd edn. Benjamin, Elmsford (1995)
2. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Anal. Mach. Intell. **24** (2002)
3. Bhat, P., Zitnick, C.L., Cohen, M., Curless, B.: Gradientshop: a gradient-domain optimization framework for image and video filtering. ACM Trans. Graph. **29** (2010)
4. Chen, T., Cheng, M.M., Tan, P., Shamir, A., Hu, S.M.: Sketch2photo: internet image montage. ACM Trans. Graph. **28** (2009)
5. Cheng, M.M., Zhang, G.X., Mitra, N.J., Huang, X., Hu, S.M.: Global contrast based salient region detection. In: IEEE CVPR (2011)
6. Coelho, F., Ribeiro, C.: Dpikt: automatic illustration system for media content. In: Proceedings of the 9th International Workshop on Content-based Multimedia Indexing (2011)
7. Collomosse, J.P., McNeill, G., Qian, Y.: Storyboard sketches for content based video retrieval. In: IEEE 12th International Conference on Computer Vision (2009)
8. Curtis, C.J., Anderson, S.E., Seims, J.E., Fleischer, K.W., Salesin, D.H.: Computer-generated watercolor. In: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques (1997)
9. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: ideas, influences, and trends of the new age. ACM Comput. Surv. **39** (2006)
10. Delgado, D., Magalhaes, J., Correia, N.: Automated illustration of news stories. In: Proceedings of the 2010 IEEE Fourth International Conference on Semantic Computing (2010)
11. Ding, M., Tong, R.F.: Content-aware copying and pasting in images. Vis. Comput. **26** (2010)
12. Eitz, M., Hildebrand, K., Boubekeur, T., Alexa, M.: Photosketch: a sketch based image query and compositing system. In: SIGGRAPH 2009: Talks, SIGGRAPH '09 (2009)
13. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. Int. J. Comput. Vis. **59** (2004)
14. Georgescu, B., Shimshoni, I., Meer, P.: Mean shift based clustering in high dimensions: a texture classification example (2003)
15. Hanser, E., Kevitt, P.M., Lunney, T., Condell, J.: Scenemaker: automatic visualisation of screenplays. In: Proceedings of the 32nd Annual German Conference on Advances in Artificial Intelligence (2009)
16. Hertzmann, A., Zorin, D.: Illustrating smooth surfaces. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (2000)
17. Jia, J., Sun, J., Tang, C.K., Shum, H.Y.: Drag-and-drop pasting. ACM Trans. Graph. (2006). SIGGRAPH
18. Jin, W., Hujun, B., Weihua, Z., Qunsheng, P., Yingqing, X.: Automatic image-based pencil sketch rendering. J. Comput. Sci. Technol. **17** (2002)
19. Johnson, M., Brostow, G.J., Shotton, J., Arandjelovic, O., Kwatra, V., Cipolla, R.: Semantic photo synthesis. Comput. Graph. Forum **25** (2006)
20. Joshi, D., Wang, J.Z., Li, J.: The story picturing engine—a system for automatic text illustration. ACM Trans. Multimedia Comput. Commun. Appl. **2** (2006)
21. Lalonde, J.F., Hoiem, D., Efros, A.A., Rother, C., Winn, J., Criminisi, A.: Photo clip art. ACM Trans. Graph. **26**(3) (2007). SIGGRAPH 2007
22. Levin, A., Lischinski, D., Weiss, Y.: A closed-form solution to natural image matting. IEEE Trans. Pattern Anal. Mach. Intell. **30** (2008)
23. Liu, Z.Q., Leung, K.M.: Script visualization (scriptviz): a smart system that makes writing fun. Soft Comput. **10** (2006)
24. Madden, M., Chung, P.W.H., Dawson, C.W.: The effect of a computer-based cartooning tool on children's cartoons and written stories. Comput. Educ. **51** (2008)
25. Manjunath, B., Ma, W.: Texture features for browsing and retrieval of image data. IEEE Trans. Pattern Anal. Mach. Intell. **18** (1996)
26. Oliva, A., Torralba, A.: Building the gist of a scene: the role of global image features in recognition. In: Progress in Brain Research, p. 2006 (2006)
27. Rother, C., Kolmogorov, V., Blake, A.: "grabcut": interactive foreground extraction using iterated graph cuts. ACM Trans. Graph. **23** (2004)
28. Schwarz, K., Rojtberg, P., Caspar, J., Gurevych, I., Goesele, M., Lensch, H.P.A.: Text-to-video: story illustration from online photo collections. In: Proceedings of the 14th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems: Part IV (2010)
29. Sekine, S.: Corpus-based parsing and sublanguage studies. Ph.D. thesis, New York University (1998)
30. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE Trans. Pattern Anal. Mach. Intell. **22** (2000)
31. Wang, C., Li, Z., Zhang, L.: Mindfinder: image search by interactive sketching and tagging. In: Proceedings of the 19th International Conference on World Wide Web (2010)
32. Wang, J.Z., Li, J., Wiederhold, G.: Simplicity: Semantics-sensitive integrated matching for picture libraries. IEEE Trans. Pattern Anal. Mach. Intell. **23** (2001)
33. Zhao, M., Zhu, S.C.: Customizing painterly rendering styles using stroke processes. In: Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Non-Photorealistic Animation and Rendering (2011)

Jiawan Zhang is currently a professor in the school of computer software and adjunct professor in the school of computer science and technology, Tianjin University. He received his Master and Ph.D. degrees in computer science from Tianjin University in 2001 and 2004, respectively. His main research interests are computer graphics and realistic image synthesis.



Yukun Hao received the B.S. degree in the School of Computer Science and Technology from Tianjin University, P.R. China, in 2010. He is currently working toward the M.S. degree in computer application at the MTS Laboratory, Tianjin University. His main research interests include image processing and information visualization.

Liang Li received the B.S. degree in the School of Computer Science and Technology from Dalian University of Technology, P.R. China, in 2008. He is currently working toward the Ph.D. degree in computer application at the MTS Laboratory, Tianjin University. His main research interests include computer vision and information visualization.

Di Sun received the B.S. degree in the School of Computer Science and Technology from Northeast Normal University, P.R. China, in 2009. She has just received the M.S. degree in computer application from Tianjin University. Her main research interests include image processing and information visualization.

Li Yuan received the B.S. degree in the School of Computer Science and Technology from Tianjin University in 2009. She has just received the M.S. degree in computer application from Tianjin University. Her main research interests include image processing and information visualization.