Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Visual knowledge guided intelligent generation of Chinese seal carving*

Kejun ZHANG[†1,2], Rui ZHANG[†1], Yehang YIN[1], Yifei LI[3], Wenqi WU[1],
Lingyun SUN[1,2], Fei WU[1], Huanghuang DENG[1], Yunhe PAN[†‡1]

[1]*College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China*
[2]*Alibaba-Zhejiang University Joint Institute of Frontier Technologies, Hangzhou 310027, China*
[3]*School of Software Technology, Zhejiang University, Hangzhou 310027, China*
[†]E-mail: zhangkejun@zju.edu.cn; zhang_rui@zju.edu.cn; panyh@zju.edu.cn

Received Feb. 22, 2021; Revision accepted June 20, 2021; Crosschecked Jan. 25, 2022; Published online Apr. 25, 2022

Abstract: We digitally reproduce the process of resource collaboration, design creation, and visual presentation of Chinese seal-carving art. We develop an intelligent seal-carving art-generation system (Zhejiang University Intelligent Seal-Carving System, http://www.next.zju.edu.cn/seal/; the website of the seal-carving search and layout system is http://www.next.zju.edu.cn/seal/search_app/) to deal with the difficulty in using a visual knowledge guided computational art approach. The knowledge base in this study is the Qiushi Seal-Carving Database, which consists of open datasets of images of seal characters and seal stamps. We propose a seal character generation method based on visual knowledge, guided by the database and expertise. Furthermore, to create the layout of the seal, we propose a deformation algorithm to adjust the seal characters and calculate layout parameters from the database and knowledge to achieve an intelligent structure. Experimental results show that this method and system can effectively deal with the difficulties in the generation of seal carving. Our work provides theoretical and applied references for the rebirth and innovation of seal-carving art.

Key words: Seal-carving; Intelligent generation; Deep learning; Parametric modeling; Computational art
https://doi.org/10.1631/FITEE.2100094 **CLC number:** TP301.6

1 Introduction

Seal carving is the art of engraving Chinese characters on seals, and it has an about 3000-year history. The seal was initially used as a practical tool that performs credibility authentication in political and economic activities. The discovery and popularization of ophicalcite enabled the literati to self-seal and carve. Due to the rise of seals for signing on art collections, the function of seal carving began to change from applied art to artistic appreciation. Over time, it became a popular form of art among the literati (Gu, 2013).

The art of seal carving adheres to classical conventions, and the seal script is the major script style used for seal carving. The correctness of the characters' orthography is traditionally an important aspect of the seal-carving technique (Li GT and Ma, 2009), and the commonly used seal script typologies are very different from modern Chinese characters. Chinese characters' structural and stroke features have changed so much that the seal script can no longer be considered a style of modern Chinese characters (Wang L, 1980). Seal script information processing technology is slowly evolving, including industry standards for encoding (Unicode Consortium, 2020), recognition, and glyph production. Seal script

‡ Corresponding author
* Project supported by the Natural Science Foundation of Zhejiang Province, China (No. LZ19F020002) and the Key R&D Program of Zhejiang Province, China (No. 2022C03126)
ORCID: Kejun ZHANG, https://orcid.org/0000-0002-0778-2303; Yunhe PAN, https://orcid.org/0000-0002-0608-3826

can still be seen in historical locations, cultural artifacts, and antiques, as well as in books and calligraphy works (Qiu et al., 2000). Previous studies of Chinese seal-carving art have primarily focused on identification and authentication of the entire seal (Fan and Tsai, 1984; Chen, 1995, 1996; Su, 2007a, 2007b), but there have been few studies on the production of seal characters. Compared with Chinese character calligraphy and other art forms, seal-carving art is time-consuming, economically costly, and laborious. Therefore, exploring the intelligent generation of seal-carving art could improve the efficiency and quality of seal-carving art creation and assist in reviving this ancient art form. It would also lower the cost of seal carving, enhance the quality of seal products, and make seal-carving art more accessible to people.

The intelligent generation of seal-carving art is complex and challenging, requiring multiple disciplines across computer science, such as data science, computer vision, computer graphics, and human-computer interaction. However, "visual knowledge" (Pan, 2019), a new form of knowledge representation, can guide many tasks on computer vision, including the generation of seal carving. In this study, we propose a method to generate a seal with a specific style and appropriate layout from a simple format of a standard script. The database of seal characters and seal stamps offers essential visual knowledge, which guides the generation of seal characters and the layout of a seal. Fig. 1 depicts the flow chart of the intelligent generation of seal carving. Our study can also be applied to other forms of art and design of Chinese characters, thereby serving the cultural and creative industry, as well as inheriting and promoting traditional culture.

2 Related works

2.1 Generation of characters

2.1.1 Based on interaction

Most of the early studies conducted on character generation were based on interactive methods. Artists undertook this type of study, and computers offer assistance in improving human design and artistic creation efficiency (Zhang JS, 2019). According to study content, it can be divided roughly into two types, Chinese calligraphy generation and font design.

Researchers study mainly the digital modeling of calligraphy tools for Chinese calligraphy generation, including virtual brush modeling and ink diffusion simulation. Wang YG and Pang (1986)

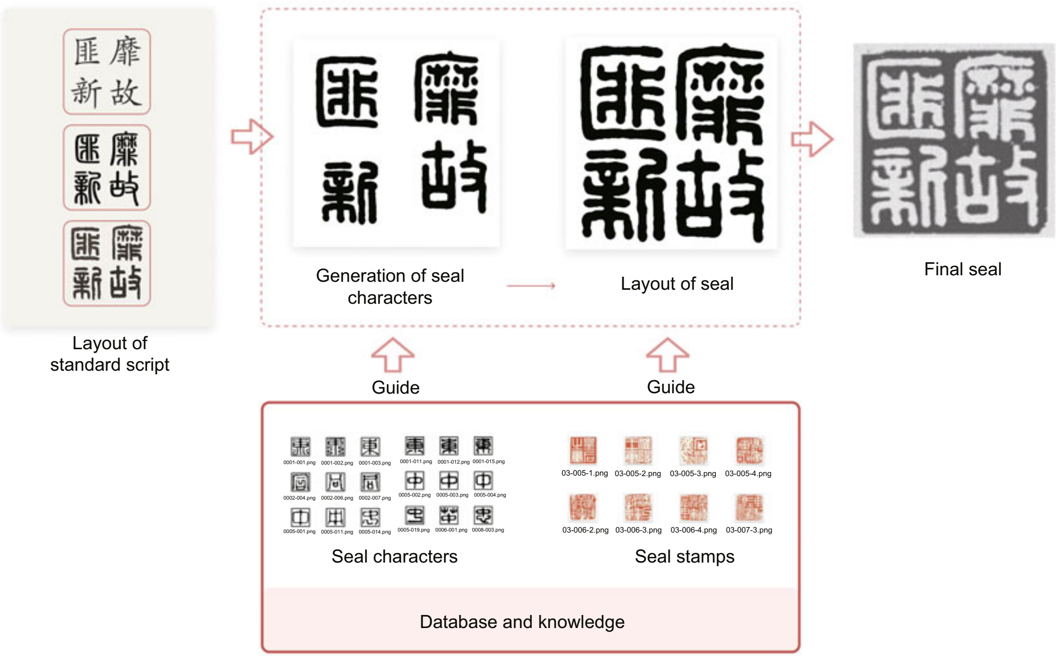

Fig. 1 Flowchart of intelligent generation of Chinese seal carving

proposed a computer-based Chinese calligraphy system. They simulated writing brushes, built a library of strokes, and used human–computer interaction to combine characters. Strassmann (1986) studied the virtual brush modeling. The author divided the functions of the brush-writing process into the brush, stroke, dip, and paper, and parameterized them separately. After that, more researchers adopted various methods of modeling. For virtual brush modeling, some researchers started from the contact shape of the brush and paper, and employed the scattered point set (Yu JH et al., 1996), ellipse (Wong and Ip, 2000), raindrop (Mi et al., 2002; Bi et al., 2003), etc. The user specified the crucial locations for the brush's movement trajectory to make calligraphic characters; however, the parameter settings are extremely complex. Xu SH et al. (2002) and Girshick (2004) developed a generation model based on the movement of the mouse. Lu et al. (2013) proposed a data-driven calligraphy and painting system, RealBrush, which can synthesize more colorful and texture effects. Some researchers also applied mechanical methods to model brushes from a physical viewpoint (Lee, 1999; Saito and Nakajima, 1999; Baxter et al., 2001). In addition, Chu and Tai (2004, 2005) installed sensors on a brush to simulate the movement of the brush in a three-dimensional space. For the ink diffusion effect, Guo and Kunii (1991) considered the paper fiber structure to simulate the dynamic diffusion of ink, and Lee (1999) proposed a Chinese art paper model using a network structure and an ink diffusion algorithm.

Many font designers choose auxiliary design systems for font creation, such as FontCreator (High-Logic, 2020), FontLab (FontLab, 2020), and Glyphs (Glyphs, 2020). Chinese font design companies, such as Founder, have also launched an auxiliary system (Founder Group, 2020) dedicated to Chinese font design. Professional designers often use these tools in font creation and adjustment.

In summary, interactive glyph generation methods can generate glyphs in a relatively controllable manner. Users can control the creation process, and the resulting glyphs are guaranteed to be straightforward and pleasing (Wang YG and Pang, 1986). However, the disadvantage of this method is that it is not intelligent enough; it depends heavily on the user's professional knowledge and still requires a lot of user work.

2.1.2 Based on graphics

Glyph generation based on graphical methods generally starts at the font component level (such as radicals, strokes, and skeletons). Then, it automatically generates glyphs through various techniques, such as parameterized representation, component mapping, and statistical models (Zhang JS, 2019). Moreover, the objects of glyph generation are mostly modern characters, such as regular, cursive, running scripts, and even personalized handwriting. Unfortunately, there are very few studies of glyph generation of seal script.

Xu SH et al. (2007, 2009) introduced a calligraphy creation method based on analogical and integrated reasoning. A matching model based on strokes was created using a hierarchical parameter representation of characters. Furthermore, the specific process disassembles the characters to be generated and obtains a new character; the matching model chooses similar strokes to match the corresponding topological structure. Shi et al. (2014) modeled Chinese character components, constructed a dynamic Bayes model, and adjusted character generation by applying the condition equation. Based on texture mapping, Yu JH and Peng (2005) proposed a method for generating cursive script and used Markov interpolation for texture synthesis. Dong et al. (2008) considered a calligraphy simulation based on statistical models. Li W et al. (2014) proposed a weighted histogram of forces to measure a character's topological features and then to synthesize topologically consistent characters.

There is significant research on personalized fonts, in addition to Chinese calligraphy. Zhou et al. (2011) proposed a model based on mapping between familiar characters and handwriting, and all characters were generated using 20% handwriting. Lian and Xiao (2012) generated handwriting by creating standard word templates, matching handwriting with familiar characters, and replacing corresponding parts. Additionally, Lin et al. (2014, 2015) developed a system for generating partial handwriting. Zong and Zhu (2014) created a component mapping vocabulary that automatically matches standard characters and handwriting. Then, without any structural input from individuals, they developed handwriting in a similar style. In summary, the generation of glyphs using graphical approaches is dependent mainly on the parametric representation and modeling of font components; the created characters are usually stable. Although visual methods can eventually develop intelligent characters, specific expert knowledge is required to disassemble and model characters. Furthermore, the number of characters is limited due to the complexity of disassembly and modeling, and the cost of large-scale use is prohibitively high.

2.1.3 Based on machine learning

In recent years, machine learning, especially deep learning, has been rapidly developed and extensively applied in computer vision, graphics, etc. For example, numerous researchers have used deep learning methods for font generation and achieved a series of results (Zhang JS, 2019). The key techniques are convolutional neural networks (CNNs), variational auto-encoders (VAEs), generative adversarial networks (GANs), recurrent neural networks (RNNs), and feed-forward neural networks (FFNNs).

Tian (2016) used CNNs to transfer font style of a regular script, built a style transfer network using five-layer convolution, and tested fonts such as song and regular scripts, but the results were poor. After "pix2pix" (a kind of GAN) (Isola et al., 2017) was proposed, Tian (2017) conducted experiments on a regular script, named "zi2zi," and good results were achieved. Jiang et al. (2017) proposed DCFont based on a deep convolutional GAN that learns to generate handwritten characters. Then they proposed SCFont (Jiang et al., 2019) using high-level information, such as skeletons and strokes. Wen et al. (2019) used CNNs and GANs to generate characters before optimizing them. To encode content and style separately and to generate Chinese characters, Zhang YX et al. (2018) used an encoder-mixer-decoder (EMD) framework. Sun et al. (2017) used a style-aware VAE (SA-VAE) to encode content and style separately, as well as multi-level information such as structures and radicals. Furthermore, using a hierarchical content generator and discriminator, Chang J et al. (2018) employed a hierarchical GAN known as the hierarchical adversarial network (HAN) to generate characters. Lyu et al. (2017) used an auto-encoding GAN to generate ancient calligraphers' handwriting. Lian et al. (2018) proposed EasyFont, which extracts and classifies strokes before using an FFNN to generate characters. Tang et al. (2019) proposed FontRNN to generate characters using an encoder and decoder as the RNN input. Chang B et al. (2018) used the unpaired CycleGAN method to generate characters, employing two generators and two discriminators. Zheng ZZ and Zhang (2018) proposed CocoANN, which uses an adversarial approach to optimize two content and style encoders to generate characters. Details are shown in Table 1.

Three data dimensions—amount, paired or not, and data label—can be used to describe the attributes of the seal character database. The amount of data is reflected mainly in the difference between the type of font and the number of characters in a single font. The more font types there are, the more generalized the algorithm will be; the smaller the amount of data, the stronger the creativity of the algorithm. Furthermore, the goal of the character-generation algorithm research is to use fewer data to obtain better results on more font types. Paired data mean that there is a one-to-one correspondence between the target and reference fonts. The machine can have a more accurate understanding when data are input in pairs into the model. The data label indicates that the algorithm uses information in addition to the image information. Algorithms that do not use additional labels use the glyph image information for training on a single font each time; to learn the associations and differences between different font categories, algorithms that use font category labels are trained on multiple fonts concurrently. Algorithms for high-level semantic tags, such as radicals, strokes, skeletons, key points, and time series of critical issues, can better understand the composition of words and achieve more accurate results. There are various models used in glyph-generation research. GANs and CNNs are the most used ones. In addition, the generator+discriminator framework and the encoder+decoder framework are most common, and multi-model combination and multi-frame nesting are usually adopted.

In summary, font-generation algorithms based on machine learning are more intelligent and do not need any expert knowledge. However, few studies have focused on seal carving. This is because seal script used in seal carving has existed for about 3000 years and is very different from standard writings. Because of this uniqueness, most of research cannot be directly applied to seal script.

Table 1 References about the generation of Chinese characters

Research	Method	Database	Additional label	Model(s)
Tian (2016)	Rewrite	3000 characters, paired data	None	CNN
Lyu et al. (2017)	AEGN	3000 characters, paired data	None	GAN
Chang J et al. (2018)	HAN	3000 characters, paired data	None	GAN
Wen et al. (2019)	CSR	750 characters, paired data	None	CNN, GAN
Chang B et al. (2018)	CycleGAN	Multiple fonts (each with 3000 characters), unpaired data	None	GAN
Zheng and Zhang (2018)	CocoANN	Multiple fonts (each with 6763 characters), unpaired data	None	CNN, GAN
Tian (2017)	zi2zi	Multiple fonts (each with 3000 characters), paired data	Font category	GAN
Jiang et al. (2017)	DCFont	Multiple fonts (each with 775 characters), paired data	Font category	GAN
Zhang YX et al. (2018)	EMD	Multiple fonts (each with 1732 characters), paired data	Font category	CNN
Sun et al. (2017)	SA-VAE	Multiple fonts (each with 3000 characters), paired data	Radical structure	VAE
Jiang et al. (2019)	SCFont	Multiple fonts (each with 775 characters), paired data	Stroke category	CNN, GAN
Lian et al. (2018)	EasyFont	Multiple fonts (each with 775 characters), paired data	Key points of skeleton	FFNN
Tang et al. (2019)	FontRNN	Multiple fonts (each with 775 characters), key points	Time series of key points	CNN, RNN

CNN: convolutional neural network; EMD: encoder-mixer-decoder; FFNN: feed-forward neural network; GAN: generative adversarial network; HAN: hierarchical adversarial network; VAE: variational auto-encoder; SA-VAE: style-aware VAE

2.2 Layout of characters

There is little research or system development concerning the layout of calligraphy or seals. Leung (2004) analyzed and synthesized traditional Chinese seals, and proposed a method for generating seal images from handwritten Chinese characters. Xu YX (2007) designed and developed an interactive calligraphy plaque-generation system that allows users to search for images of calligraphy characters with consistent content from the established calligraphy database; users can adjust the layout of characters on the plaque. Yu K (2010) designed a system that can calculate the layout based on the size of the plaque and unify the size and stroke width of calligraphy characters. However, related studies on the layout of characters on seals are rare.

3 Method

"Visual knowledge" (Pan, 2019) is a new form of knowledge representation that can guide many visual tasks, for example, transformation from one visual image to another. Pan (2021) took intelligent generation of Chinese seal carving as a typical example, and we will show how visual knowledge guides us.

Visual knowledge K can express the dimension, color, texture, spatial shape, and spatial relationships of an object. The task of style transfer can be expressed as $K^Y = G(K^X)$, where K^X is the knowledge of style X, K^Y is the knowledge of style Y, and G is the generator that transforms the styles from X to Y. In our task, we take the visual knowledge of seal carving as K_{SC}. Style X is the layout of the standard script and style Y is the real seal. A feasible way to achieve $K_{\text{SC}}^Y = G(K_{\text{SC}}^X)$ is obtaining paired data of seals with styles X and Y and training a deep learning model. However, this simple method does not effectively use visual knowledge. For example, some sub-knowledge between styles X and Y is the same, such as the character skeletons on the seals. Some sub-knowledge can also be obtained using graphical and statistical methods, such as layout of characters. The division and explicit expression of this sub-knowledge can guide and improve the generation significantly. There are many different glyphs of the same characters, and they can hardly be paired on the pixel level, which means that the end-to-end model can barely work.

Thus, we propose a novel method guided by visual knowledge. A visual concept usually has a hierarchical structure. For the visual knowledge of seal carving K_{SC}, we construct a hierarchical structure as shown in Fig. 2. Now we have the layout of the standard script K_{SC}^X, and our target is to obtain a novel seal K_{SC}^Y. The character set of the standard script is sufficient, but the set of characters used in seal carving is incomplete. Thus, we can use all knowledge of style X, i.e., K^X, as the source rather than style Y. The standard script is written by brush, whereas the characters on the seals are carved, so their textures are different but the skeletons are almost the same. Thus, we assume that the skeleton knowledge between styles X and Y is the same, i.e., $K_{\mathrm{SK}}^X = K_{\mathrm{SK}}^Y$. Then we need only to obtain K_{SC}^Y from K_{SK}^Y.

In detail, to obtain a seal carving K_{SC}^Y, we construct a system to obtain a seal stamp K_{S}^Y and combine it with a carving machine K_{Ca}. To obtain a seal stamp K_{S}^Y, we generate seal characters K_{C}^Y and calculate the layout K_{L}^Y from our seal stamp dataset. In addition, we propose a deformation algorithm to combine K_{C}^Y and K_{L}^Y. Then we construct a paired dataset of the incomplete characters K_{C}^Y and their skeletons, and train a deep learning model to obtain rendering K_{R}^Y. Now we need only to obtain $K_{\mathrm{SK}}^Y = K_{\mathrm{SK}}^X$, which can be easily extracted from standard scripts K_{C}^X.

4 Seal-carving database and knowledge

Visual learning of seals is the prominent issue for intelligent generation of Chinese seal carving, so we construct the Qiushi Seal-Carving Database as the knowledge base. The visual knowledge of seal stamps K_{S} has two parts, seal character K_{C} and layout K_{L}. To obtain the layout's knowledge K_{L}^Y, we construct the seal stamp dataset, and to obtain the seal character's knowledge K_{C}^X and K_{C}^Y (incomplete), we build the seal character dataset. We combine computer and manual methods to obtain the data and compile them by different kinds of styles. There are about 6500 seal stamps and 70 000 seal characters in the Qiushi database, which is our knowledge base for generation and layout.

First, we collect seal stamp books such as *Hanyin Fenyun Hebian* (Yuan, 1979), *Zhongguo Lidai Yinfeng* (Huang, 1999), and *Zhuanke Changyongzi Zidian* (Liu, 2010). After collecting a large amount of data, we use an intelligent seal-carving image-processing procedure to construct the database, as depicted in Fig. 3. After scanning, page segmentation, deskew, matching, single-character segmentation, style clustering, standardization, text-image matching, and index creation, the seal book is finally processed into standard data classified by the style. Because seals are often old and their edges are easily worn, it is usually not a complete square when it is stamped on the paper. For zhuwen (characters in red), the boundary of the seal is often not obvious. We use pixel-level calculations supplemented by professional knowledge to segment each seal image and the corresponding text in the seal book. Moreover, we correct the rotation of the stamp using the Hough transform, i.e., deskew. Generally, the seal image contains multiple characters, and to facilitate generation research, we separate the characters from images using a crowdsourcing platform. This study applies optical character recognition, supplemented with manual checking, to obtain text-annotation images for matching between the text and single-character image. Considering the various

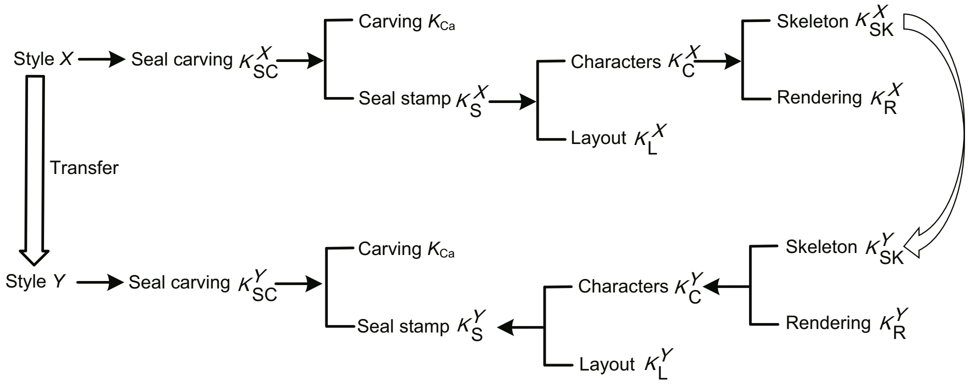

Fig. 2 Visual knowledge of seal carving

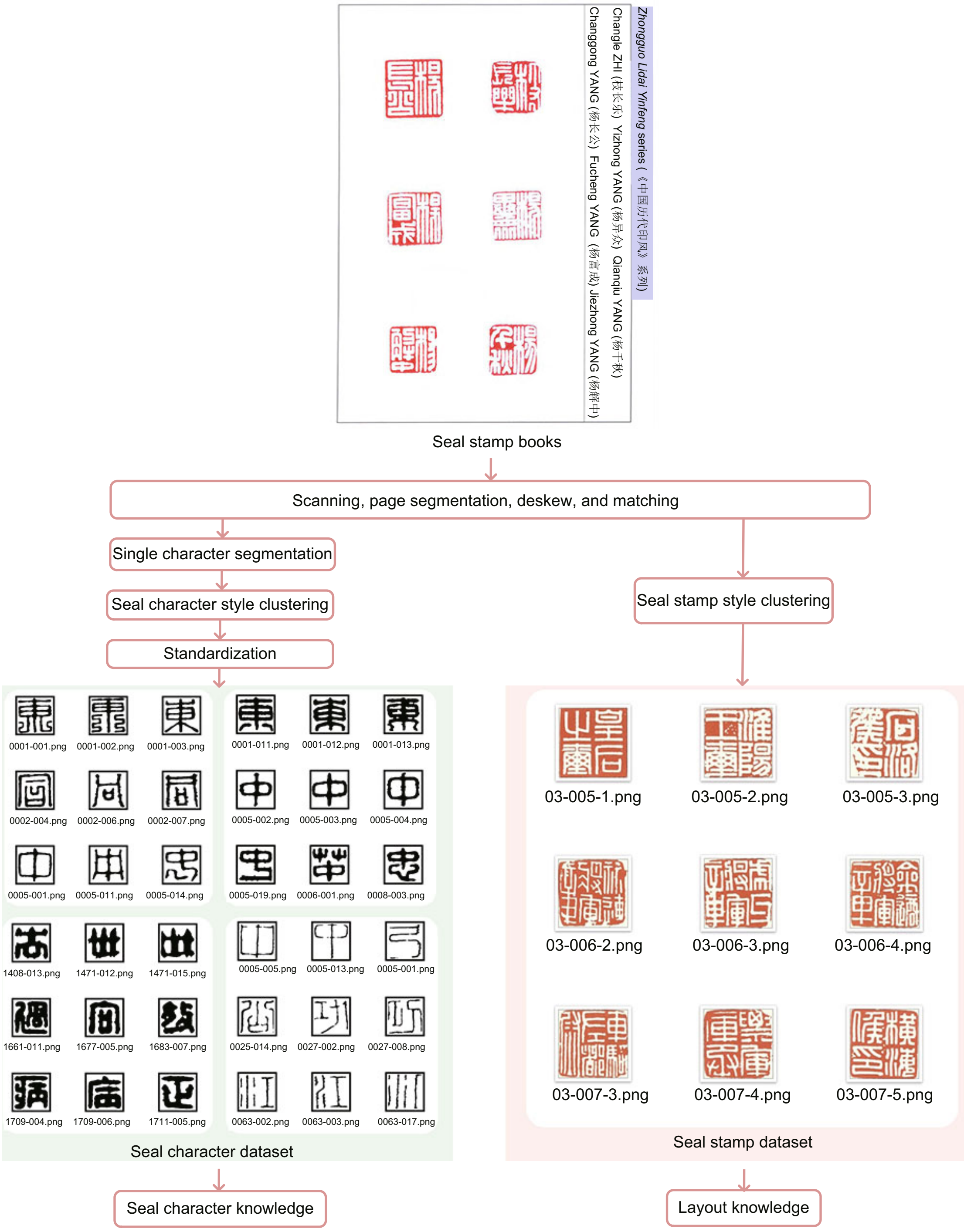

Fig. 3 Construction of the seal-carving database (References to color refer to the online version of this figure)

seal-carving styles, the data are sorted according to different styles to facilitate further research. We use some key style features, such as thickness, stroke angle distribution, and vertical and horizontal symmetry. We achieve good results, such as zhuwen and baiwen (characters in white) binary classification and style clustering. In the same style, the characters in the two-character seal images are long, but they are square in the four-character seal images. Furthermore, each character has a different layout range on the sealing surface because of the number of strokes, so the size of each character differs. Therefore, a standardized method is needed to make the characters within the same style class have consistent style features and the same size. If we stretch the characters directly, they will be deformed, and their appearance and even style will be affected. This study applies medial axis transformation to divide the characters into skeletons and distance distributions, and then they are separately adjusted to make the size and thickness uniform.

In addition, in this study, we supplement the seal-carving database itself. The data-driven method of deep learning is applied to intelligently generate seal characters and seals. After selecting high-quality results and adding them to the seal-carving database, the database grows continuously. The designed standard database is deployed on the seal character retrieval platform. Additionally, users can input a single character to be retrieved, and the system will return the seal image and seal character image with the character (Fig. 4).

Fig. 4 Seal glyph retrieving platform

5 Generation of seal characters

In this section, we will show how we obtain the target characters K_{C}^{Y}. Although we obtain some target characters on the seal from our database, the character set lacks many characters because there are some characters not shown in the existing seal stamp books. Thus, we use these target characters to learn the rendering K_{R}^{Y} by a deep learning model. Combining skeletons K_{SK}^{X} from the complete character set K_{C}^{X} and the target rendering K_{R}^{Y}, we obtain all the target characters K_{C}^{Y}.

In practice, we set seal characters on seals from *Zhongguo Lidai Yinfeng* (Huang, 1999) as style Y, called the style of Hanyin. Then we set characters in the Dictionary of Common Characters for Seal Carving (Liu, 2010) as style X. The skeletons of Miu seal characters in the Dictionary of Common Characters for Seal Carving are correct and artistic, but the characters are written with a brush, which is not suitable for seal. As shown in Fig. 5, during training, we extract the skeleton of seal characters on seals and combine the skeletons K_{SK}^{Y} and characters K_{C}^{Y} as pairs. Then we input the pairs into a GAN model named zi2zi (Tian, 2017). After the training, the model can learn how to render a skeleton into a character with the Hanyin style. During generation, we extract the skeletons K_{SK}^{X} of the Miu seal characters in the Dictionary of Common Characters for Seal Carving, and input the skeletons into the model to generate the characters. After artifact removal, the characters can be used in the seal. Our method divides the skeleton knowledge and rendering and develops the characters with skeletons of Miu seal characters and Hanyin style.

This method maintains the skeleton of the original Miu seal script style. It thus avoids structural errors and fusion of neighboring strokes, which are universal in end-to-end models. The brush-written texture is removed on the generated glyphs and therefore fits well into seal-carving aesthetics. The skeleton-based generation method, either the seal script or the modern Chinese characters, is chosen as the training data. The model's inference can result in a corresponding style of generated seal glyphs. For example, if we want a "harder" style for seal characters, we can employ modern Chinese characters in the Gothic font style as style Y for training. The shortcomings of this method are the lack of structural changes and the dependence of the generated character set on the source dataset.

6 Layout

To obtain seal stamps K_{S}^{Y} from the layout of seal K_{L}^{Y}, we first propose a deformation algorithm to adjust the seal character size and location while maintaining the character shape. Using this deformation algorithm, we can compose the characters with interaction. Then, to make the layout more intelligent, we calculate layout parameters from the database to achieve a smart layout.

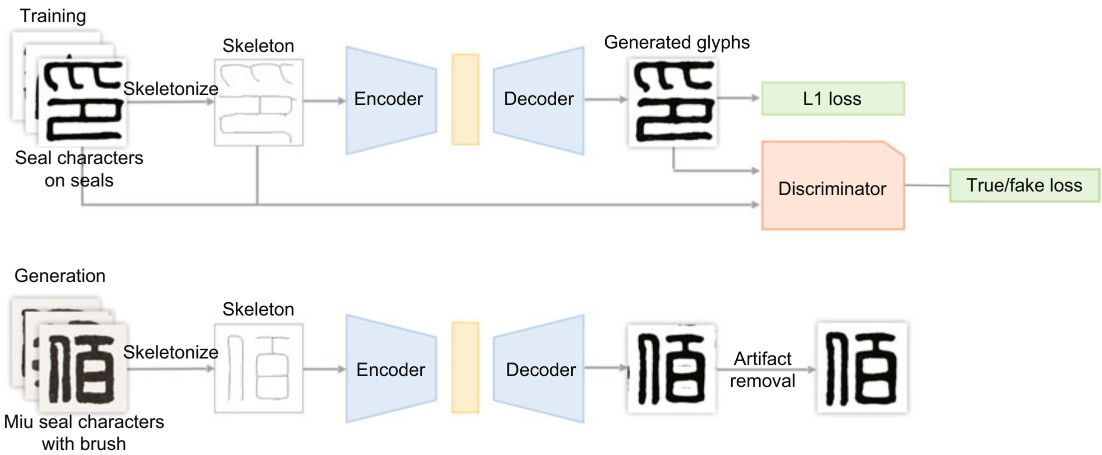

Fig. 5 Generation model based on the skeleton guided by hierarchical visual knowledge

6.1 Deformation algorithm

Based on mathematical morphology and vector parameterization, we propose an intelligent deformation algorithm as a basis for character layout. Different from pictures, if scaling and other deformations are simply applied to the glyphs, the strokes will be out of shape and the structural characteristics will be changed. For example, if you want to compress the Chinese glyphs horizontally, the glyphs will become narrower, and the vertical drawing should become thinner to maintain symmetry. Therefore, simple scale processing at the pixel level is impossible.

In this study, the method based on mathematical morphology uses mainly skeletonization and medial axis transformation. The binarized Chinese character image is converted into a skeleton image (denoted by $\boldsymbol{S}$, which is a matrix with the same size as the original image) and the distribution map (denoted by $\boldsymbol{D}$, which is a matrix with the same size as the original image) of the nearest distance from each point in the skeleton image to the glyph outline. The converted $\boldsymbol{S}$ and $\boldsymbol{D}$ can be restored to the original image by applying a simple image algorithm: traverse all the skeleton points on $\boldsymbol{S}$, for each point, obtain the corresponding value in $\boldsymbol{D}$ named d, and draw a circle on this point with a radius of d; the superposition of all the circles is the original image. The manipulation of $\boldsymbol{S}$ and $\boldsymbol{D}$ can cause the structure and stroke shape of the corresponding glyph image to change, respectively. For example, if you want to change the font size without changing the thickness, you can scale both $\boldsymbol{S}$ and $\boldsymbol{D}$ to the target size (with the same multiplicator), and then restore both $\boldsymbol{S}$ and $\boldsymbol{D}$ to a font image. Moreover, if you want to thicken or thin the font stroke, you can multiply $\boldsymbol{D}$ by a value while keeping $\boldsymbol{S}$ constant, and then restore $\boldsymbol{S}$ and $\boldsymbol{D}$ to a font image.

The mathematical morphology based glyph manipulation method needs to recalculate the skeleton and distribution maps of the closest distance many times, so its computational cost is very high. Therefore, we propose a vector parameterized deformation method that has high storage efficiency and image transmission speed in browser/server (B/S) system applications. Moreover, this method can be regarded as the result of the previous method after sampling on the image contour. Particularly, polygons are used to fit the contour of the image polygon, the nodes of the polygon contour are recorded, and the skeleton points closest to these nodes are recorded. As shown in Fig. 6, we label the polygon contour node set as $C = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_n\}$, and its corresponding skeleton point is labeled as $P = \{\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_n\}$; we construct the offset vector of the node on the skeleton point $O = \{\boldsymbol{o}_i | \boldsymbol{o}_i = \boldsymbol{c}_i - \boldsymbol{s}_i,\ i = 1, 2, \ldots, n\}$. Thus, the manipulation of P and O can lead to changes in the restored polygon outline node set. The sequence of nodes and their contours will be saved in advance, and the set of contour nodes can always be restored to the glyph image. For example, if you employ this method to change the font size without changing the thickness, you can scale P to the target size without changing O. Another example is to thicken or thin the font stroke by multiplying O by a value.

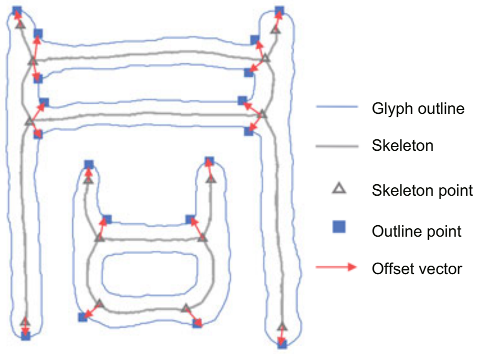

Fig. 6 Parameterized model based on hierarchical visual knowledge

In this study, we develop an intelligent layout system that enables users to manipulate character deformation and layout in real time and preview the results based on the intelligent deformation algorithm. As illustrated in Fig. 7, users can first query the corresponding seal characters, search in our developed seal-carving database, add them to the layout surface, and then drag the vertices of the characters' bounding boxes to adjust sizes and positions of the characters on a seal in real time. To ease the user operations, the interactive interface also provides an automatic character layout based on the average layout method. We obtain the default parameters by applying statistical techniques. The characters can be quickly and evenly arranged on the layout surface, and they will be easier to fine-tune based on this arrangement. On one hand, using the developed intelligent interaction can generate complex and diverse layouts of seal-engraving prints, thereby avoiding unreasonable character layouts. On the other hand, for complex glyphs that do not exist in some database, users can split them into two or multiple simple glyphs and join them together to make complex glyphs.

6.2 Intelligent layout guided by database and knowledge

In this subsection, we calculate layout parameters K_{L}^{Y} from the database and knowledge. After that, we apply these parameters to the layout, and the appropriate layout is achieved.

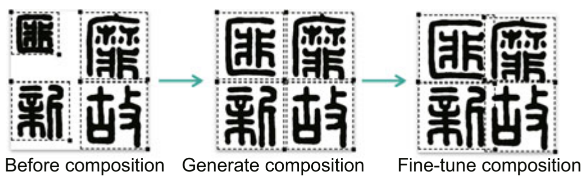

Fig. 7 User interface for seal composition

The first parameter is the thickness of the stroke. By the parameterized model in Fig. 6, we can calculate the norm of the offset vector, which means the thickness of the stroke. After calculating the thickness of all characters with the same style in the seal stamp dataset, we can obtain the average thickness of this style. Then we adjust the offset vector's norm without changing the direction. Finally, the characters on the seal have the same thickness of strokes as the chosen styles.

Two parameters determine the location and size of the characters: margin and character spacing. First, we calculate the convex hull of all the characters in the seal (i.e., the yellow lines in Fig. 8). Second, we calculate the hull of each character in the seal separately (i.e., the blue lines in Fig. 8). Then we calculate the media axis of the four hulls (i.e., the red lines in Fig. 8). The average of twice the distance between the convex hull and the media axis of the seal can represent the margin. The average space between the hull of one character and the media axis or the convex hull of all characters represents the character spacing of this character. By calculating the average character spacing of all characters, we can obtain the final character spacing.

Hanyin usually adopts an average layout, so the media axes of four characters are determined. After calculating the layout parameters above, we can obtain the coordinates of the four points of each character's bounding box. Then we use the deformation algorithm to change the characters' size and complete the final layout.

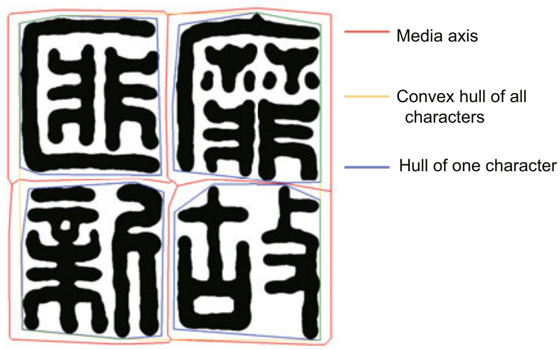

Fig. 8 Calculation of margin and character spacing (References to color refer to the online version of this figure)

7 Experiments

7.1 Dataset and experimental settings

To evaluate the ability to generate seal carving, we used seal characters on seals from *Zhongguo Lidai Yinfeng* (Huang, 1999) as target style Y, called the style of Hanyin. We used 3485 seal stamps in total. Then we used layouts of characters in the Dictionary of Common Characters for Seal Carving (Liu, 2010) as style X. Finally, we used 6030 characters and combined them into layouts that were paired with the actual seal stamps.

We chose the pix2pix (Isola et al., 2017) model as the baseline model, and used a data augmentation method to develop more than 9000 paired images. We trained the pix2pix model for 200 epochs.

We randomly selected nine seal stamps as the testing set and compared the results of both methods. Because there are no ground truth images for comparison, we recruited many participants to evaluate the results.

7.2 Experimental results

As shown in Fig. 9, our method is better than the GAN. We obtained more stable and more aesthetically pleasing results. There were too many structural errors and fusion of neighboring strokes in the GAN results. The same character can have many different glyphs, and they are difficult to pair at the pixel level for the pix2pix model. Another reason is that character layouts on the seal are not simple, making it hard to match characters on the seal and characters of the standard script. Some seals generated by our method are not good enough, such as the first one. The character in the bottom right corner is too large, and although we have a parameter to adjust this, the default result is not good enough. The seventh one is not good either, compared with the true seals.

To evaluate these results, we conducted a user study to compare these three kinds of results. To ensure that participants did not know the seal's style, we shuffled the seals. There were 31 non-specialists (Experiment I) and 20 professionals (Experiment II) involved, and we did the statistical analysis separately. We selected several metrics for non-specialists (I): aesthetic harmony, visual balance, texture roughness, stroke spacing, stroke uniformity, and overall aesthetics. For professionals (II), we added two metrics: variation of strokes and space distribution. Every metric has 1–7 points that can be chosen, and the higher the point, the better the performance.

The descriptive statistics is provided in Table 2.

To learn more about the differences between the three kinds of seals on these metrics, we first determined the homogeneity of variances and then conducted the ANOVA test and robust equality of means tests. For $p < 0.05$, we used Welch and for $p > 0.05$, we used the ANOVA test. As shown in Table 3, the results indicated a significant difference between these three kinds of seals.

Then we conducted multiple comparisons as shown in Table 4. For non-specialists, our results are better than those of the GAN and the true seals. Our results are also better than those of the GAN for professionals, but there are no significant differences between our results and the true seals.

Fig. 9 Experiment results (GAN: generative adversarial network)

Separately comparing the 27 seals in Fig. 9, non-specialists believe that our first and fifth seals are worse than the true seals while the second and fourth seals are better. However, professionals think that our first, seventh, and ninth seals are worse, while the second, third, and fourth ones are better.

In summary, our method is better than the GAN without a doubt. Non-specialists think that our seals are better than true seals, and the professionals believe that our seals and true seals have no significant differences.

8 System

After preliminary exploration of the intelligent generation of seal-carving art, we integrated the steps of construction and retrieval of the seal-carving database, smart generation of seal characters, smart layout of characters on the seal, and seal carving to design an integrated intelligent seal-carving system, as depicted in Fig. 10. The system was launched in June 2020, and the "AI Seal-Carving Experience Activity" was organized. The system was used by school students to generate their seals.

Given that users are mostly non-specialists and that the operations such as printing are inconvenient to perform on mobile terminals, our team adopted a more intelligent and easy-to-use method to design the system. The process involves three stages: customizing the seal characters and style, adjusting the seal interactively, and waiting for carving. At the first stage, the users type the content text and select the corresponding style. The system extracts the related seal characters from our database and uses the average layout method to obtain the preliminary results. At the second stage, borders, stroke thickness, rounded corners, character structures, margins, etc., can be adjusted. At the last stage, the user submits the seal carving and waits for the seal-carving machine to complete the seal.

To combine the seal stamps with the sealing machine, we used ArtCAM to convert the image to toolpaths used for numerical control engraving (Yin et al., 2020). Each generated seal stamp was saved as an image and uploaded to the server-side. The upper computer of the numerical control (CNC) engraving system then downloaded the image of the seal and converted it to toolpaths in G-code. The conversion

Table 2 Descriptive statistics

Experiment	Seals	Aesthetic harmony	Visual balance	Texture roughness	Stroke spacing	Stroke uniformity	Variation of strokes	Space distribution	Overall aesthetics
I	GAN	2.574	2.746	2.850	2.358	2.366			2.459
	Ours	5.520	5.448	5.057	5.480	5.326			5.548
	True seals	4.986	5.086	4.642	4.957	4.953			5.022
II	GAN	4.150	3.978	4.383	4.006	3.983	4.011	3.922	4.000
	Ours	5.106	5.161	4.933	5.372	5.417	5.072	5.067	5.461
	True seals	5.117	5.250	4.983	5.267	5.267	5.044	5.028	5.267

Table 3 ANOVA analysis results

Experiment	Aesthetic harmony	Visual balance	Texture roughness	Stroke spacing	Stroke uniformity	Variation of strokes	Space distribution	Overall aesthetics
I	687.464*	600.434*	214.835*	686.366*	533.84*			749.338*
II	31.533*	90.802*	9.336*	75.105*	71.685*	77.405*	38.356*	43.332*

$^{*}p < 0.001$

Table 4 Multiple comparisons

Experiment	Seals	Aesthetic harmony	Visual balance	Texture roughness	Stroke spacing	Stroke uniformity	Variation of strokes	Space distribution	Overall aesthetics
I	Ours-GAN	2.946**	2.706**	2.208**	3.122**	2.961**			3.090**
	Ours-true seals	0.534**	0.362*	0.416*	0.523**	0.372*			0.527**
II	Ours-GAN	0.956**	1.183**	0.550**	1.367**	1.433**	1.061**	1.144**	1.461**
	Ours-true seals	−0.011	−0.089	−0.050	0.106	0.150	0.028	0.039	0.194

$^{*}p < 0.01$; $^{**}p < 0.001$

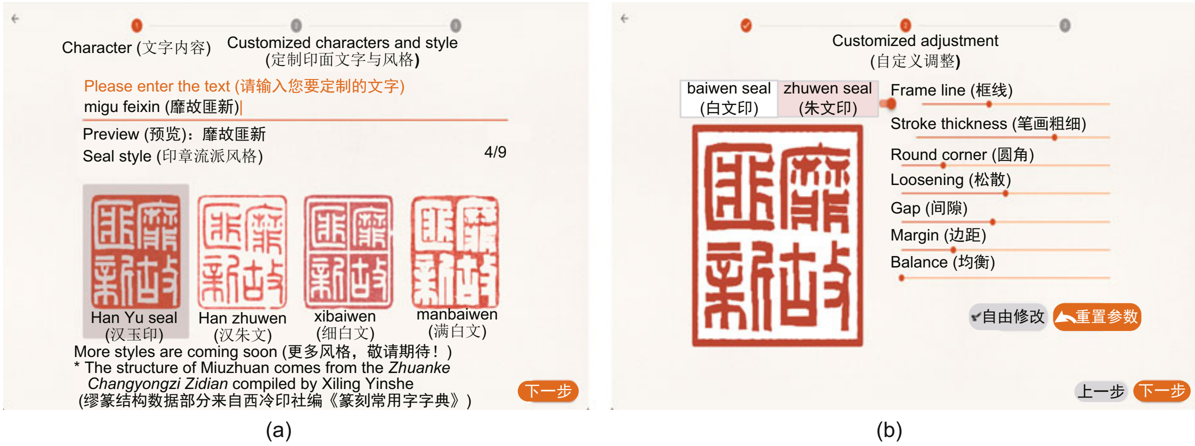

Fig. 10 Integrated intelligent system for seal-carving art generation: (a) the first step, customizing the seal characters and style; (b) the second step, adjusting the seal interactively

was done by ArtCAM, a popular computer-aided manufacturing tool. With the G-code instructions sent from the upper computer via the USB serial port, the CNC engraving machine started to engrave the seal design on a stone, under the control of the lower programmable logic controller in the system.

As an attempt to promote the culture of seal carving to the public and allow users to experience the art of seal carving, this "AI Seal-Carving Experience Activity" achieved great success and provided a reference for cultural development in the field of Chinese character art. Moreover, as an essential research object in Chinese characters, seal carving has significant cultural and academic value. As a product of ancient symbols and modern computer science and technology, the intelligent integrated seal-carving system demonstrates the auxiliary ability of computer science in the study of art.

9 Conclusions

Chinese seal carving, an ancient traditional culture, deserves to be carried forward. The intelligent generation of Chinese seal carving is a critical study that will improve the efficiency and quality of seal-carving art creation and will make seal carving more accessible to people. In this paper, we proposed a pipeline for the intelligent generation of Chinese seal carving guided by visual knowledge. First, we constructed the Qiushi Seal-Carving Database, including the seal character dataset and seal stamp dataset, and extracted seal character knowledge and layout knowledge from it. Then, guided by the seal character knowledge, we proposed a generation method for seal characters. Guided by hierarchical visual understanding, we proposed a deformation algorithm. With the help of the layout knowledge and deformation algorithm, we achieved an intelligent layout. Finally, a layout of a standard seal script can become a usable seal after the generation of seal characters and layout. In this study, we provided a reference for computer-art intelligent generation and support for the study of Chinese characters, especially for ancient characters. On one hand, clear goals and requirements may impose new conditions on technology, and current mainstream technologies may face challenges. On the other hand, technological innovation may certainly bring new experience to aesthetics and promote the inheritance and spread of culture. The perfect combination of technology and art is our ultimate goal.

There is also much weakness of our work. For example, although the use of skeletons improves stability, it also restricts the variety of seals. A possible way to resolve this is to train a new model to transfer the skeleton between different styles. Now we have excellent results on the Hanyin style, but many styles are valuable to generate. To create great seal styles, we need to expand our datasets and use visual knowledge more effectively.

Contributors

Kejun ZHANG, Rui ZHANG, and Yunhe PAN designed the research. Rui ZHANG, Yehang YIN, and Yifei LI processed the data. Kejun ZHANG and Rui ZHANG drafted the paper. Yehang YIN, Yifei LI, Wenqi WU, Lingyun SUN, Fei WU, Huanghuang DENG, and Yunhe PAN helped organize the paper. Kejun ZHANG and Rui ZHANG revised and finalized the paper.

Compliance with ethics guidelines

Kejun ZHANG, Rui ZHANG, Yehang YIN, Yifei LI, Wenqi WU, Lingyun SUN, Fei WU, Huanghuang DENG, and Yunhe PAN declare that they have no conflict of interest.

References

Baxter B, Scheib V, Lin MC, et al., 2001. DAB: interactive haptic painting with 3D virtual brushes. Proc 28th Annual Conf on Computer Graphics and Interactive Techniques, p.461-468. https://doi.org/10.1145/383259.383313

Bi XF, Tang M, Lin JZ, et al., 2003. An experience based virtual brush model. *J Comput Res Dev*, 40(8):1244-1251 (in Chinese).

Chang B, Zhang Q, Pan SY, et al., 2018. Generating handwritten Chinese characters using CycleGAN. IEEE Winter Conf on Applications of Computer Vision, p.199-207. https://doi.org/10.1109/WACV.2018.00028

Chang J, Gu YJ, Zhang Y, et al., 2018. Chinese handwriting imitation with hierarchical generative adversarial network. British Machine Vision Conf, Article 290.

Chen YS, 1995. Computer processing on the identification of a Chinese seal image. Proc 3rd Int Conf on Document Analysis and Recognition, p.422-425. https://doi.org/10.1109/ICDAR.1995.599027

Chen YS, 1996. Automatic identification for a Chinese seal image. *Patt Recogn*, 29(11):1807-1820. https://doi.org/10.1016/0031-3203(96)00032-5

Chu NSH, Tai CL, 2004. Real-time painting with an expressive virtual Chinese brush. *IEEE Comput Graph Appl*, 24(5):76-85. https://doi.org/10.1109/MCG.2004.37

Chu NSH, Tai CL, 2005. MoXi: real-time ink dispersion in absorbent paper. *ACM Trans Graph*, 24(3):504-511. https://doi.org/10.1145/1186822.1073221

Dong J, Xu M, Pan YH, 2008. Statistic model-based simulation on calligraphy creation. *Chin J Comput*, 31(7):1276-1282 (in Chinese). https://doi.org/10.3321/j.issn:0254-4164.2008.07.023

Fan TJ, Tsai WH, 1984. Automatic Chinese seal identification. *Comput Vis Graph Image Process*, 25(3):311-330. https://doi.org/10.1016/0734-189X(84)90198-1

FontLab, 2020. FontLab 7. https://www.fontlab.com/ [Accessed on Jan. 1, 2021].

Founder Group, 2020. FounderType. http://www.foundertype.com/ [Accessed on Jan. 1, 2021].

Girshick RB, 2004. Simulating Chinese brush painting: the parametric hairy brush. ACM SIGGRAPH, Article 22. https://doi.org/10.1145/1186415.1186442

Glyphs, 2020. Glyphs. https://glyphsapp.com/ [Accessed on Jan. 1, 2021].

Gu ZL, 2013. Eight Lectures on the Basis of Seal Cutting. Shanghai Painting and Calligraphy Publishing House, China (in Chinese).

Guo QL, Kunii TL, 1991. Modeling the diffuse paintings of 'Sumie'. In: Kunii TL (Ed.), Modeling in Computer Graphics. IFIP Series on Computer Graphics. Springer, Tokyo. https://doi.org/10.1007/978-4-431-68147-2_21

High-Logic, 2020. High-Logic. https://www.high-logic.com/ [Accessed on Jan. 1, 2021].

Huang D, 1999. Zhongguo Lidai Yinfeng. Chongqing Press, Chongqing, China (in Chinese).

Isola P, Zhu JY, Zhou TH, et al., 2017. Image-to-image translation with conditional adversarial networks. IEEE Conf on Computer Vision and Pattern Recognition, p.5967-5976. https://doi.org/10.1109/CVPR.2017.632

Jiang Y, Lian ZH, Tang YM, et al., 2017. DCFont: an end-to-end deep Chinese font generation system. SIGGRAPH, Article 22. https://doi.org/10.1145/3145749.3149440

Jiang Y, Lian ZH, Tang YM, et al., 2019. SCFont: structure-guided Chinese font generation via deep stacked networks. Proc AAAI Conf on Artificial Intelligence, p.4015-4022. https://doi.org/10.1609/aaai.v33i01.33014015

Lee J, 1999. Simulating oriental black-ink painting. *IEEE Comput Graph Appl*, 19(3):74-81. https://doi.org/10.1109/38.761553

Leung H, 2004. Analysis of traditional Chinese seals and synthesis of personalized seals. IEEE Int Conf on Multimedia and Expo, p.1283-1286. https://doi.org/10.1109/ICME.2004.1394458

Li GT, Ma SD, 2009. Zhuankexue. Jiangsu Education Publishing House, China (in Chinese).

Li W, Song YP, Zhou CL, 2014. Computationally evaluating and synthesizing Chinese calligraphy. *Neurocomputing*, 135:299-305. https://doi.org/10.1016/j.neucom.2013.12.013

Lian ZH, Xiao JG, 2012. Automatic shape morphing for Chinese characters. SIGGRAPH, p.1-4. https://doi.org/10.1145/2407746.2407748

Lian ZH, Zhao B, Chen XD, et al., 2018. EasyFont: a style learning-based system to easily build your large-scale handwriting fonts. *ACM Trans Graph*, 38(1):6. https://doi.org/10.1145/3213767

Lin JW, Wang CY, Ting CL, et al., 2014. Font generation of personal handwritten Chinese characters. Proc 5th Int Conf on Graphic and Image Processing, Article 90691T. https://doi.org/10.1117/12.2050128

Lin JW, Hong CY, Chang RI, et al., 2015. Complete font generation of Chinese characters in personal handwriting style. IEEE 34th Int Performance Computing and Communications Conf, p.1-5. https://doi.org/10.1109/PCCC.2015.7410321

Liu J, 2010. Zhuanke Changyongzi Zidian. Xiling Yinshe, China (in Chinese).

Lu JW, Barnes C, DiVerdi S, et al., 2013. RealBrush: painting with examples of physical media. *ACM Trans Graph*, 32(4):117. https://doi.org/10.1145/2461912.2461998

Lyu PY, Bai X, Yao C, et al., 2017. Auto-encoder guided GAN for Chinese calligraphy synthesis. 14th IAPR Int Conf on Document Analysis and Recognition, p.1095-1100. https://doi.org/10.1109/ICDAR.2017.181

Mi XF, Xu J, Tang M, et al., 2002. The droplet virtual brush for Chinese calligraphic character modeling. Proc 6th IEEE Workshop on Applications of Computer Vision, p.330-334. https://doi.org/10.1109/ACV.2002.1182203

Pan YH, 2019. On visual knowledge. *Front Inform Technol Electron Eng*, 20(8):1021-1025. https://doi.org/10.1631/FITEE.1910001

Pan YH, 2021. Miniaturized five fundamental issues about visual knowledge. *Front Inform Technol Electron Eng*, 22(5):615-618. https://doi.org/10.1631/FITEE.2040000

Qiu X, Matto GL, Norman J, 2000. Chinese Writing. Institute of East Asian Studies, University of California, USA.

Saito S, Nakajima M, 1999. 3D physics-based brush model for painting. ACM SIGGRAPH, Article 226. https://doi.org/10.1145/311625.312110

Shi C, Xiao JG, Xu CH, et al., 2014. Automatic generation of Chinese character using features fusion from calligraphy and font. The Engineering Reality of Virtual Reality, Article 90120N. https://doi.org/10.1117/12.2038945

Strassmann S, 1986. Hairy brushes. *ACM SIGGRAPH Comput Graph*, 20(4):225-232. https://doi.org/10.1145/15886.15911

Su CL, 2007a. Edge distance and gray level extraction and orientation invariant transform for Chinese seal recognition. *Appl Math Comput*, 193(2):325-334. https://doi.org/10.1016/j.amc.2007.03.061

Su CL, 2007b. Ring-to-line mapping and orientation invariant transform for Chinese seal character recognition. *Int J Comput Math*, 84(1):11-22. https://doi.org/10.1080/00207160701197967

Sun DY, Ren TZ, Li CX, et al., 2017. Learning to write stylized Chinese characters by reading a handful of examples. https://arxiv.org/abs/1712.06424v3

Tang SS, Xia ZQ, Lian ZH, et al., 2019. FontRNN: generating large-scale Chinese fonts via recurrent neural network. *Comput Graph Forum*, 38(7):567-577. https://doi.org/10.1111/cgf.13861

Tian YC, 2016. Rewrite. https://github.com/kaonashi-tyc/Rewrite [Accessed on Jan. 1, 2021].

Tian YC, 2017. zi2zi. https://github.com/kaonashi-tyc/zi2zi [Accessed on Jan. 1, 2021].

Unicode Consortium, 2020. Roadmap to the TIP. https://unicode.org/roadmaps/tip/ [Accessed on Jan. 1, 2021].

Wang L, 1980. Manuscript of Chinese History. Zhong Hua Book Company, Beijing, China (in Chinese).

Wang YG, Pang YJ, 1986. CCC—computer Chinese calligraphy system. *Inform Contr*, (2):38-43 (in Chinese).

Wen C, Chang J, Zhang Y, et al., 2019. Handwritten Chinese font generation with collaborative stroke refinement. https://arxiv.org/abs/1904.13268

Wong HTF, Ip HHS, 2000. Virtual brush: a model-based synthesis of Chinese calligraphy. *Comput Graph*, 24(1):99-113. https://doi.org/10.1016/S0097-8493(99)00141-7

Xu SH, Tang M, Lau F, et al., 2002. A solid model based virtual hairy brush. *Comput Graph Forum*, 21(3):299-308. https://doi.org/10.1111/1467-8659.00589

Xu SH, Jiang H, Lau FCM, 2007. An intelligent system for Chinese calligraphy. Proc 22^{nd} National Conf on Artificial Intelligence, p.1578-1583.

Xu SH, Jiang H, Jin T, et al., 2009. Automatic generation of Chinese calligraphic writings with style imitation. *IEEE Intell Syst*, 24(2):44-53. https://doi.org/10.1109/MIS.2009.23

Xu YX, 2007. The Research and Implementation of Chinese Calligraphy Tablet Generation Techniques. MS Thesis, Zhejiang University, Hangzhou, China (in Chinese).

Yin YH, Chen ZW, Zhao YJ, et al., 2020. Automated Chinese seal carving art creation with AI assistance. IEEE Conf on Multimedia Information Processing and Retrieval, p.394-395. https://doi.org/10.1109/MIPR49039.2020.00086

Yu JH, Peng QS, 2005. Realistic synthesis of *cao shu* of Chinese calligraphy. *Comput Graph*, 29(1):145-153. https://doi.org/10.1016/j.cag.2004.11.013

Yu JH, Zhang JD, Cong YQ, 1996. A physically-based brush-pen model. *J Comput-Aid Des Comput Graph*, 8(4):241-245 (in Chinese). https://doi.org/10.3321/j.issn:1003-9775.1996.04.001

Yu K, 2010. Researches on Some Key Technologies of Computer Calligraphy. PhD Thesis, Zhejiang University, Hangzhou, China (in Chinese).

Yuan R, 1979. Hanyin Fenyun Hebian. Shanghai Bookstore Publishing House, China (in Chinese).

Zhang JS, 2019. A survey of digital calligraphy. *Sci Sin Inform*, 49(2):143-158 (in Chinese).

Zhang YX, Zhang Y, Cai WB, 2018. Separating style and content for generalized style transfer. IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.8447-8455. https://doi.org/10.1109/CVPR.2018.00881

Zheng ZZ, Zhang FY, 2018. Coconditional autoencoding adversarial networks for Chinese font feature learning. https://arxiv.org/abs/1812.04451

Zhou BY, Wang WH, Chen ZH, 2011. Easy generation of personal Chinese handwritten fonts. IEEE Int Conf on Multimedia and Expo, p.1-6. https://doi.org/10.1109/ICME.2011.6011892

Zong A, Zhu YK, 2014. StrokeBank: automating personalized Chinese handwriting generation. Proc 28^{th} AAAI Conf on Artificial Intelligence, p.3024-3029.