Interactive Animation Editing Based on Sketch Interaction

Yan Huang[1,2(✉)], Ti Zhou[2], Yanfeng Li[1], Yan Zhang[1],
and Cuixia Ma[2,3]

[1] Nanjing Software Research Institute, Nanjing, China
huangyanll4@mails.ucas.ac.cn
[2] University of Chinese Academy of Sciences, Beijing, China
cuixia@iscas.ac.cn
[3] Institute of Software, Chinese Academy of Sciences, Beijing, China

Abstract. User-centric interactive system shows potential in allowing users facilely access to the experience of natural interaction appropriate to user's intention with a low cognitive load. A novel approach, central to the user's experience, is presented based on knowledge reuse, user interaction model and sketch-based interaction. A proposed sketch platform for animation editing provide the key to instantiating typical application within the reuse methods. The sketch-based interface explores a point in the tradeoff between expressiveness and naturalness to provide users a natural interactive environment. It helps to fill in the gaps of traditional WIMP (Window Icon Menu Pointing Device) pattern in the process animation drawings by sketching, exploring and modifying their ideas interactively with immediate and continuous visual feedback, and validate existing efforts and provide impetus for future work in the area of natural interaction research.

Keywords: Knowledge reuse · User-centric · Sketch · Natural user interface

1 Introduction

The booming of portable devices and networks helps to the development of innovative interaction pattern and operation behavior in the area of human-computer interaction. The design thinking of man-machine interface is being migrated from the pattern research of graphical interface to user-centric interactive design [1]. For improving the efficiency of knowledge utilization on a user-centric interface, a lack of an available method for knowledge induction, sorting, archiving may make a contribution to the poor reuse of existing data in the process of user interactive design. Since knowledge reuse, with its guidance, foreseeability and reappearance, is instructive for the interface design and interaction theory. In this paper, taking animation cases as application background, we make the most of knowledge reuse to present a sketch animation editing method that supports user with simple, natural sketch drawing on operating and organizing animation appropriate to user's intention with a low cognitive load. As animations are ubiquitous in modern visual media, the animation editing is an interactive, visual prototype with an interface resembling a sketch editor or drawing

application, which provides an open work surface similar to the user's sketching lines. The user can sketch static illustrations or outline annotations directly on the objective animation using a simple sketch operation. The editing metaphor is extended with tools for sketching, interactive visual elements on the animation surface. Considering that user's editing stroke represents the timeline of an animation that can be used to achieve any operation like cutting, splicing, selecting or modification, We take advantage of the ability of sketching to provide efficient interaction directly to perform knowledge reuse for user design with the animation elements in order to make it easier and faster to work with animation editing.

The main contributions of this paper include: (1) Analysis the interactive behavior of user-computer to provide a sketch-based user model, which provides available theoretical guidance and assessment principles for the natural sketch editing method. (2) Demonstrate a sketch modeling of the multi-layered solution mechanism to achieve sketch structural recognition and understanding. (3) Provide a sketch-based animation editing platform that supports knowledge reuse for user facilely access to animation design in a natural environment. The work we conducted shows that it enhances the accessibility and individualization during the process of user drawing design and helps to enhance user experience in human computer interaction.

2 Related Work

Animation design is one of the most challenging problem-solving activities that the modern system faces. The cognitive complexity of design problems and the respond-time pressure due to improved interactive efficiency force designers to pursue efficient processes and strategies [2]. Natural editing method have become the central feature of user intention expression and are becoming the more fundamental and critical part in human interaction [3]. The traditional freehand writing on a paper is that simple and provides good user experience during the knowledge understanding. Most current animation designs for interactive interface are intuitive and multi-style which present different, rich menu items for users accessing operations. But they cannot support the interactively creative ideas description and achieve natural interaction well, which is important for user experience in the process of communicating with system. With the popularity of computers and pen interaction technologies, ideas of new methods and interactive technologies are incited to aid the sketch design and application. We prefer combining technologies of sketch-based recognition and understanding to explore an effective knowledge reuse in natural animation editing system. Some related techniques are involved, such as cognitive model, knowledge reuse and sketching technology.

Cognitive Model. Since the user tasks in the systems are becoming more complex, the benefit of systematic interface design receives more emphasis [4]. A good interface design procedure should elicit the user's task needs and task knowledge, which to some extent, leads to the demand of user cognitive factor on the special interactive interface [2]. To establish effective design strategies, the practical cognition of different users' needs to be considered [5]. Exploring cognition mechanism of user behavior helps to

illustrate the user cognition model in the calculation process of user analysis, modeling and decision-making [6] when they interact with the interface. To realize natural and efficient interaction, we need to build the mapping mechanism to make computer understand the human cognitive process and shorten the cognitive distance between themselves thus reducing the burden of users' cognition. In this paper, we explore the user behavior and characteristics in the cognitive process to provide a sketch user model.

Knowledge Reuse. Natural interface design is becoming a more complex task due to the increasing needs of functionality and convenience of animation editing systems. Since User's constrained cognitive capability has made natural interface challenge further, we should optimize the interaction between the user and interface considering user goals, cognitive capabilities and available interface intention in the process of animation editing. Thus the concept of knowledge reuse has been proposed to minimize many redundant works and save the time cost for user editing and drawing [7, 8]. Among the existing approaches to achieve systematic knowledge reuse, some may appropriately focus on the preservation of design rationale for the issue-based information systems [9–11], some may consider the component-based software development with reusable modularized components practiced to depict object-oriented modelling techniques [12, 13], and others may prefer case-based reasoning techniques to find a prior problem similar to the current problem and provide a modified solution [14, 15]. This techniques have been applied to various design domains such as architecture, mechanical or physical devices and software [16–18]. In this paper, we probe the opportunity to achieve better user design by presenting relevant design cases based on sketch-related method. The sketch knowledge representation is therefore taken as the most important success factor, for which we propose organization of cases for users efficiently access to the animation design.

Sketching Interface. Sketch-based interaction presents a new generation natural user experience that breaks the bottleneck of current WIMP pattern and adopt the nature and efficiency of pencil and paper based interaction. Freehand sketching is a natural and crucial part of everyday human interaction, especially in early design processing [5]. Sketching technique mimics the tradition pen and paper that represents a natural way of thinking about ideas, its interface satisfies users in a freeform and flexible manner, allowing them to use their pen to draw sketches actively when interacting with computer systems [19]. With sketch interaction and recognition method, we provide user feedbacks to improve and enhance the learning process. By creating sketch-based interface, users can editing data more easily while keeping a natural style of interaction [20, 21]. Sketching technology has become popular for interactively design in different domains, like animation [22, 23], garment design, mathematics sketching, botanical modeling [24], floral modeling [25], hair style modeling [26, 27] and so on. In these systems, sketching can be used to create the prototype of design quickly. However, in our work, recognizing and understanding of sketches user depicting and designing can be performed based on the animation design.

3   Knowledge Reuse for Adaptive Sketching User Interface

3.1   Sketch-Based User Interaction Model

User model aims to explore the user characteristics in the process of the interaction by grasping user's cognitive habits, predicting their behaviors, thus to provide theoretical guidance for reuse design of user interface. User plays a central role in generating knowledge information from the snippets of data emerging from visual interaction, which enables users to analyze huge information spaces in order to support complex decision making and data exploration. Exploring the process of knowledge generation based on user's cognition helps to effective modeling of user visual analysis and interaction. In this paper, we adhere to the principle of user-centered interactive characteristics and rule of sketch-based editing method, referenced to the knowledge generation process [28] to provide a sketch-oriented user interaction model (see Fig. 1), which illustrates the usual interactive process that enables users to interact with the system using sketch interaction. The model demonstrates the cyclical process of user-computer interaction, the user part illustrates the knowledge generation process of the human, which depicts a reasoning process composed of exploration, verification, and knowledge generation loops. The interaction part presents the interactive information on the sketch interface with its different context provided so that a sketch-oriented object can be developed to support various operations and applications. The contexts contained in the system includes, **User context:** experienced designers or novice. **Environment context:** devices, pen locations, pen-styles, colors, symbol, diagram etc. **Sensed context:** time, sequence, color and positioning information etc. **Interpreted context:** first level derived context from sense context: classification of texts, symbols, drawings, gestures, etc. Second level derived from first level: geometry, topological relations, and sketch constraints.
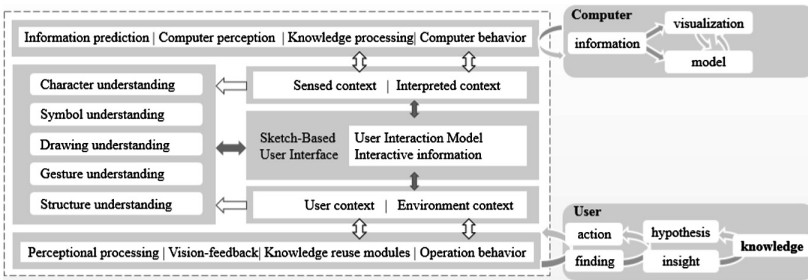

Fig. 1.   Sketch-based user interaction model

3.2   Adaptive Sketch Recognition and Understanding

Due to the various ways user depicting sketches and its intentions, uncertainties over the process of interaction make it appropriate to consider the dissimilarity among users to develop knowledge reuse sketch context and constraint [29]. Thus to achieve an adaptive sketch-based interface for facilitating the design reuse process, modeling the

sketching process is of great significance to improve the sketch recognition and understanding in the process of animation editing and interaction. We define the sketch including strokes that involves different constraints, strokes should be operated in the vector format, since a stroke is a set of sequent points captured from an input device during a time period between a pair of pen-down and pen-up events. Each point has a time stamp, with which we can recognize and compute the stroke to understanding the whole sketch structure. Admitting that different users may draw several strokes continuously or overlapping when they are required to finish a straight line, we need to group some strokes into different edges, which is a high level geometrical concept. Any modifications in either sketch strokes may contribute to the corresponding changes in another, which demonstrates the necessity of sketch modeling (Fig. 2 shows the corresponding sketch modeling).
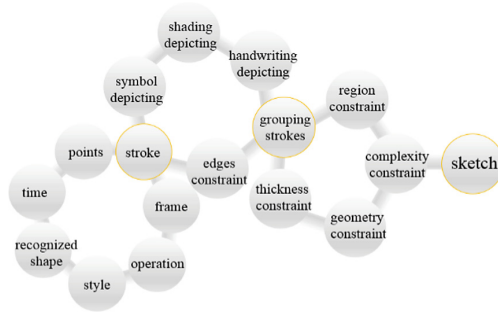
Fig. 2. The sketch modeling for recognition and understanding

To reuse the knowledge in interface design that users input the sketch data, system adaptively responds the retrieval and recognition for providing the existing similar recommendations. Adaptive recognition and understanding here means it can spontaneously adjust to variations of any sketches the user editing and provide optimal result realignment. Since the sketches users depicting presents the integrated concepts of their intents, we need to split them into every single meaningful pieces and then support a partial matching for sketch recognition [30, 31]. Figure 3 gives examples of a sketch drawings, structural recognition and understanding. For a set of sketches, detection and segmentation of pieces (nodes) should be first conducted. Considering that close distance between strokes among every piece may lead to inefficiency of recognition by simply computing the distance of strokes, we propose a new method to confirm the relative position of sketch nodes and use RNN classifier to train and recognize the feature vector of that obtained position. On the basic of recognition for connecting lines, we aim to merge the strokes according to the distance of strokes and the density of a stroke set. Denote that $dist(S_i, S_j)$ as the distance between stroke set S_i and S_j, $stroke\_thres$ as the merging threshold, while $dist(S_i, S_j) < stroke\_thres$, taking the merging operation since $stroke\_thres = \frac{3}{2n} \sum_{i=1}^{n} d_i$, where n represents the total number of strokes in a sketch, d_i represents the diagonal length of the select box that can contain

(a) Original input sketches   (b) Recognition method   (c) Result of recognition
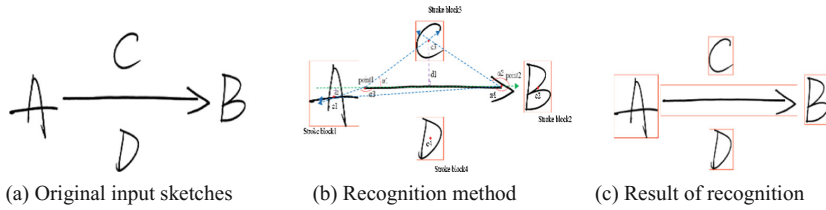
Fig. 3. An example of sketch drawings, recognition and understanding

stroke i. Meanwhile, computing the stroke density $den(B_i)$ that has been merged and its density threshold *block_thres* using $den(B_i) = \frac{\sum_{j=1}^{k} len_j}{dia_i}$, in which len_j is the length of stroke j, $j \in B_i$, dia_i *is* the diagonal length of the bounding box of stroke set B_i. If $den(B_i) < block\_thres$, abandon the merging. With the iteration of recognition for each piece of strokes, further work should be conducted to achieve the recognition of the whole sketches a user depicting. In Fig. 3, the drawing nodes are shown on the left side, the corresponding recognitions are generated by circling every piece on the right side. Its recognition can be similar to the process of stroke merging to obtain a set of stroke B_i. The central point of B_i is denoted $C_i(x_i, y_i)$, and two endpoints of the arrow, $p_1(x_1, y_1)$, $p_2(x_2, y_2)$ constitute the vector $v_1(x_1 - x_i, y_1 - y_i)$, $v_2(x_2 - x_i, y_2 - y_i)$, in which their angles α1 and α2 helps to verdict the relative position of each stroke set. If $cos\alpha1 * cos\alpha2 < 0$, we conclude that stroke sets are distributed at the top or bottom side of the arrow. If $cos\alpha1 * cos\alpha2 \geq 0$, they are distributed around the arrow. Among of which,

$$cos\alpha1 = \frac{(x_1 - x_i) \times (x_2 - x_1) + (y_1 - y_i) \times (y_2 - y_1)}{\sqrt{(x_1 - x_i)^2 + (y_1 - y_i)^2} \times \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}$$

$$cos\alpha2 = \frac{(x_2 - x_i) \times (x_2 - x_1) + (y_2 - y_i) \times (y_2 - y_1)}{\sqrt{(x_2 - x_i)^2 + (y_2 - y_i)^2} \times \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}$$
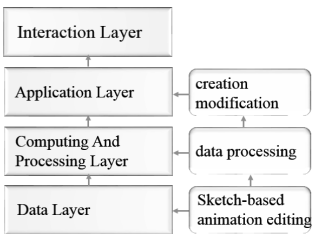

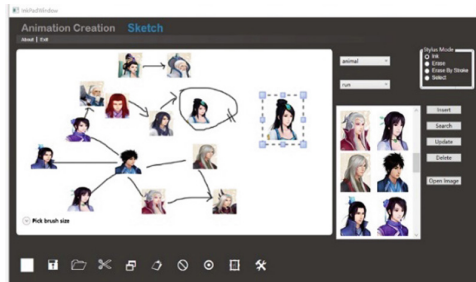**Fig. 4.** The hierarchy structure of system

Fig. 5. Example of sketch editing and processing

After obtaining the relative position of the stroke sets, we can confirm whether they belong to the arrow by judging that distance $d \leq threshold$ or $d > threshold$ thus to achieve the recognition of each piece of strokes.

3.3 Knowledge Reuse for User Interface Design

The objective of the sketch-based animation editing system is to provide users with benefits of a traditional pen-pencil tool that has taken the effective design knowledge reuse into a consideration. The design of history backtracking develops effective means of adapting to the history information reuse for it helps to amend the sketch editing at any time in history in the process of interactive design. An orderly history backtracking means that stretching the design process back to previous drawing result in the form of backtracking command. For achieving the history knowledge reuse in the sketch interface, we provide a backtracking algorithm of random concept based on user's design intent as follow.

Denote the current collection of selected objects as $P = \{\psi_1, \psi_2, \ldots, \psi_m\}$, $j, k \in (1, n)$, FirstPrev = null, SecondPrev = null, the design node of conceptual object ψ_i :<eleId, attributes, head>. ψ_i.head shows the head pointer of the constraint chain, the constraint node is defined as Cons: *<Type, ElefirstID, ElesecondID, Attributes, ConsNextforfirst, ConsNextforsecond>*, in which ConsNextforfirst points to the next constraint node of object ElefirstID, and ConsNextforsecond points to the next constraint node of ElesecondID,

- For step 1, $j = k$, if $j > n$, it goes to step 4, $i = j + 1$, the current design object is ψ_j, with the index value, the current object and its mapping constraint chain can be available.
- For step 2, searching for the constrain chain of current object, if $i > n$, $k = k + 1$, then it goes back to step 1, and taking search matching of the two design objects that the constrain nodes had recorded as $<\psi_j, \psi_i>$ ($<\psi_{i,}\psi_j>$). If it succeed, goes to step 3, or $i = i + 1$, then goes to step 2.
- For Step 3, the current constrain node is Con, FirstPrev records the previous constrain pointer or the mapping head pointer of ψ_j, SecondPrev records the previous constrain pointer or the mapping head pointer of ψ_i, FirstPrev = Con.ConsNextforfirst, SecondPrev = Con.ConsNextforsecond, then delete Con, goes to step 2.
- Step 4, end of the program.

The sketch-based backtracking in the process of drawing design helps to enhance the users' thinking process, with which users can take advantage of the ability of the reuse of history information to achieve an effective design on sketch editing. The system we provide allows users to input their own sketch knowledge to design and editing the animation, which includes animation importing and modification on the input knowledge by only using sketch interaction. The input knowledge they produce can in turn be added to the knowledge base for further reuse and reedit. If a regular user leaves, the loss will be prevented because his/her entering knowledge and experience will have already been saved for reuse. Figure 4 demonstrates the hierarchy structure of the editing system for knowledge design process, which includes four layers: data

layer, computing and processing layer, application layer and interaction layer, in which, data layer depicts the animation knowledge and experience during the inputting process, as well as the original (reuse) input animation the system had provided before. It provides general data access for the external application. Computing and processing layer provides functions to recognize and process the inputting data collected by sketching from the users. As leading to the personalities of the systemic achievement. Application layer provides functions of creation, modification, save of the animation input drawings. And interaction layer provides the flexible sketch-based interaction to improve the natural human interaction with the purpose of reducing the cognitive load of different users. Figure 5 illustrates an example of the sketching editing of the animation and their structural recognition. When designing and editing any animation using sketches, users can save the depicting into the sketchborad for their reusing and editing with the functions toolbar for modification.

4   Conclusion and Future Work

A user-centric natural interaction, has the potential to alter the way that user's usual access to the knowledge creation process with distinct advantages. In this paper we present a natural sketch-based animation editing and design based on knowledge reuse, aiming to improve the design efficiency and user experience since sketch-based interaction helps to achieve the fact that reducing the user's cognitive load. The research involved detailing and demonstrating a sketch-based user interaction model, a method of sketch modeling and a supporting knowledge reuse user interface. It is capable and applicable for modeling various sketching interfaces and knowledge modeling applications. Of course, various input constraints for sketch structure recognition and understanding can be considered as one kind of improvement for our future work. A long term goal is to explore the algorithms of extending the abilities of adaptation and recognition of complex sketches, enriching them with further user interaction model. These characters would be beneficial to improvements of more natural interactive applications, capable of addressing the limits due to different users' cognitive ability, capturing users' intentions and then performing available presentations.

References

1. Ghasemifard, N., Shamsi, M., Kenari, A.R.R., et al.: A new view at usability test methods of interfaces for human computer interaction. Glob. J. Comput. Sci. Technol. **15**(1) (2015)
2. Kim, H., Wan, C.Y.: Supporting the cognitive process of user interface design with reusable design cases. Int. J. Hum.-Comput. Stud. **62**(4), 457–486 (2005)
3. Feng, S., Wan, J.: User interface knowledge reuse and multi-device user interface development. In: IEEE International Conference on Automation and Logistics, pp. 1203–1208. IEEE (2007)

4. Kalawa, A., Occello, Derypinna, A.M., et al.: Reusing user interface across devices with different design guidelines. In: Fourth International Conference on Knowledge and Systems Engineering, pp. 211–216 (2012)

5. Ma, C., Wang, H., Teng, D., Dai, G.: Design and analysis of conceptual sketches based on context awareness. In: Lee, S., Choo, H., Ha, S., Shin, I.C. (eds.) APCHI 2008. LNCS, vol. 5068, pp. 328–335. Springer, Heidelberg (2008). doi:10.1007/978-3-540-70585-7_37

6. Fu, X.L., Cai, L.H., Liu, Y., Jia, J., Chen, W.F., Yi, Z., Zhao, G.Z., Liu, Y.J., Wu, C.X.: A computational cognition model of perception, memory, and judgment. Sci. China Inf. Sci. **57**(3), 1–15 (2014)

7. Ockerman, J.J., Mitchell, C.M.: Case-based design browser to support software reuse: theoretical structure and empirical evaluation. Int. J. Hum.-Comput. Stud. **51**, 865–893 (1999)

8. Sutcliffe, A.G., Carroll, J.M.: Designing claims for reuse in interactive systems design. Int. J. Hum.-Comput. Stud. **50**, 213–241 (1999)

9. Lee, J., Lai, K.Y.: What's in design rationale? In: Moran, T.P., Carroll, J.M. (eds.) Design Rationale: Concepts, Techniques, and Use, pp. 271–289. Lawrence Erlbaum Associates, Mahwah (1996)

10. Chen, G., Lu, R., Jin, Z.: Constructing virtual domain ontologies based on domain knowledge reuse. J. Softw. **3**, 007 (2003)

11. Pathak, A., Patra, B.K.: A knowledge reuse framework for improving novelty and diversity in recommendations. In: Proceedings of the Second ACM IKDD Conference on Data Sciences, pp. 11–19. ACM (2015)

12. Lubars, M.D., Harandi, M.T.: Knowledge-based software design using design schemas. In: Proceedings of the Ninth International Conference on Software Engineering, California, USA, pp. 253–262 (1987)

13. Sutcliffe, A.: On the effective use and reuse of HCI knowledge. ACM Trans. Comput.-Hum. Interact. (TOCHI) **7**(2), 197–221 (2000)

14. Domeshek, E.A., Kolodner, K.L.: The designers' MUSE: experience to aid conceptual design of complex artifacts. In: Maher, M.L., Pu, P. (eds.) Issues and Applications of Case-Based Reasoning in Design, pp. 11–38. Lawrence Erlbaum Associates, New York (1997)

15. Wahid, S.: Investigating design knowledge reuse for interface development. In: Conference on Designing Interactive Systems, pp. 354–356. ACM (2006)

16. Mou, Y., Cao, J., Zhang, S.: A process component model for enterprise business knowledge reuse. In: Proceedings of the 2004 IEEE International Conference on Services Computing, pp. 409–412. IEEE Computer Society (2004)

17. Cheung, P.K, Chau, P.Y.K., Au, A.K.K.: Does knowledge reuse make a creative person more creative? In: Pacific Asia Conference on Information Systems, PACIS 2005, Bangkok, Thailand, 7–10 July 2005, pp. 219–227 (2005)

18. Zelewski, S., Kowalski, M., Bergenrodt, D.: "Intelligent" knowledge reuse for complex logistics projects: an application of ontology-driven and case-based reasoning. J. Control Sci. Eng. **1**, 23–37 (2013)

19. Ma, C., Du, Y., Teng, D., et al.: An adaptive sketching user interface for education system in virtual reality. In: IEEE International Symposium on IT in Medicine & Education, ITIME, vol. 1, pp. 796–802 (2009)

20. Shesh, A., Chen, B.: Smartpaper: an interactive and user friendly sketching system. Comput. Graph. Forum **23**(3), 301 (2004)

21. Ma, C., Dai, G., Teng, D.: Gesture-based interaction computing in conceptual design. J. Softw. **16**(2), 303–308 (2005)

22. Thorne, M., Burke, D., van de Panne, M.: Motion doodles: an interface for sketching character motion. ACM Trans. Graph. **23**(3), 424–431 (2004)
23. Igarashi, T., Moscovich, T., Hughes, J.F.: Spatial keyframing for performance-driven animation. In: Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2005, pp. 107–115. ACM Press, New York (2005)
24. Okabe, M., Owada, S., Igarashi, T.: Interactive design of botanical trees using freehand sketches and example-based editing. In: Eurographics 2005 (2005)
25. Ijiri, T., Owada, S., Okabe, M., Igarashi, T.: Floral diagrams and inflorescences: interactive flower modeling using botanical structural constraints. ACM Trans. Graph. **24**(3), 720–726 (2005)
26. Mao, X., Kato, H., Imamiya, A., Anjyo, K.: Sketch interface based expressive hairstyle modeling and rendering. In: Proceedings of IEEE Computer Graphics International (CGI), pp. 608–611 (2004)
27. Malik, S.: A sketching interface for modeling and editing hairstyles. In: Eurographics Workshop on Sketch-Based Interfaces and Modeling (2005)
28. Sacha, D., Stoffel, A., Stoffel, F., et al.: Knowledge generation model for visual analytics. IEEE Trans. Vis. Comput. Graph. **20**(12), 1604–1613 (2014)
29. Davis, R.: Magic paper: sketch-understanding research. Computer **40**(9), 34–41 (2007)
30. Ma, C.-X., Liu, Y.-J., Yang, H.-Y., Teng, D.-X., Wang, H.-A., Dai, G.-Z.: KnitSketch: a sketch pad for conceptual design of 2D garment patterns. IEEE Trans. Autom. Sci. Eng. **8**(2), 431–437 (2011)
31. Bresler, M., Průša, D., Hlaváč, V.: Online recognition of sketched arrow-connected diagrams. Int. J. Doc. Anal. Recogn. (IJDAR) **19**(3), 253–267 (2016)