Neurocomputing 168 (2015) 35–46

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Lead curve detection in drawings with complex cross-points

Jia Chen [a], Min Li [c,1], Qin Jin [b,*], Shenghua Bao [d], Zhong Su [c], Yong Yu [a]

[a] Department of Computer Science and Engineering, Shanghai Jiao Tong University, China
[b] School of Information, Renmin University of China, China
[c] IBM China Research Laboratory, Beijing, China
[d] IBM Watson Group, San Jose, United States

ARTICLE INFO

Article history:
Received 27 November 2014
Received in revised form
15 April 2015
Accepted 10 June 2015
Communicated by Luming Zhang
Available online 20 June 2015

Keywords:
Lead curve
Curve-path
Cross-point graph
Structured learning

ABSTRACT

Lead curve detection in design drawings is a critical problem in a wide range of applications ranging from checking similar drawings in patent granting to constructing hyperlinks between image and text description in digitalization. The difficulty of the problem are two folds: unknown end point of lead curve and complex crossings. However, most previous curve detection algorithms are usually applied in simple or no crossing situations. We make four contributions in solving the problem: (1) we transform the problem into a new problem that finds an optimal path with the best score in the cross-point graph. We introduce the "cross-point graph" representation which captures the topology of cross-point connectedness. Based on the original drawing and the corresponding cross-point graph, we introduce the coupling concept "curve-path", which correlates the curve in the original drawing with the corresponding path in the cross-point graph. (2) we design a set of joint feature representations for curve-path which describes different characteristics of a curve and its corresponding path. (3) we define a task specific loss function for our customized structured SVM. We propose a mixed negative instance sampling strategy to learn the weights of different joint feature representations. We prune the search space effectively for fast lead curve detection. (4) we build a software to efficiently facilitate manual lead curve labeling. We release the patent drawing dataset with groundtruth to public for lead curve detection research. The extensive experimental results prove the effectiveness of our methods.

1. Introduction

Engineering design and drawings such as patent drawings, room designs and machine designs are very important types of drawings. Most of these drawings are publicly available as scanned images rather than its original Computer Aided Design (CAD) format. For example, patent drawings on the US patent site[1] are submitted in image format. The amount of such drawings increases rapidly. Take US patent for example, there were 36,034 design patent applications and 23,468 design patent grants in year 2013 [1]. This number is still increasing each year. Design patent alone has 501,790 grants during 1963 $\sim$ 2013. Numerous room designs and machine designs are also available online as scanned images.

In these drawing images, there are not only curves describing the object but also "lead curves" linking the reference characters with a certain part of the object. An example is shown in Fig. 1: there are lead curves linking reference characters such as "110", "112", "112A" to different parts of the object. These reference characters will have further detailed explanations in the document. Based on statistics of the collected patent design images in our dataset, there are around 19 lead curves on average in a patent design drawing, which indicates that lead curves frequently appear in patent drawings.

Lead curves play an important role in many drawing related applications. For example, when a drawing image of a new patent is submitted, patent agency needs to check whether it is near duplicate or similar to any drawing images of granted patents. The existence of many lead curves as shown in Fig. 1 dramatically deteriorates patent image retrieval quality. Detecting these lead curves and removing them is beneficial for near duplicate/similar design patent detection. Another example is that in the current online patent database, there is no hyperlink between the drawing image and text description. That is, users have to manually switch between the object part in the drawing image and the detailed explanation in the text, which is tedious and inefficient. Once the lead curve in the patent image is automatically detected, a hyperlink can be constructed to ease the switching effort.

Standards for Drawings [2] only provides one hint for locating the start-point of a lead curve: it must originate in the immediate

* Corresponding author.
E-mail addresses: chenjia@apex.sjtu.edu.cn (J. Chen), mushi.lm@alibaba-inc.com (M. Li), qjin@ruc.edu.cn (Q. Jin), baoshhua@us.ibm.com (S. Bao), suzhong@cn.ibm.com (Z. Su), yyu@apex.sjtu.edu.cn (Y. Yu).

[1] url http://www.uspto.gov/

http://dx.doi.org/10.1016/j.neucom.2015.06.018
0925-2312/

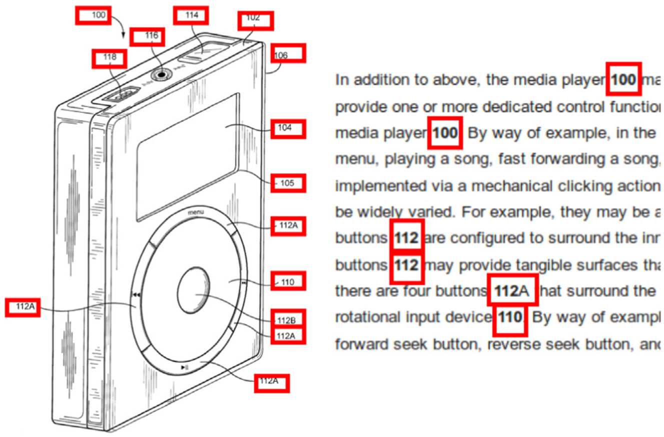

Fig. 1. An example of patent "Method and apparatus for use of rotational user inputs" (US 7345671 B2), one of Steve Jobs's top 5 favourite patents: reference characters are highlighted in red rectangles. In the drawing, lead curves link the reference characters to the object part. In the text description, there are detailed explanations for the corresponding reference characters. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

proximity of the reference character. It also adds one constraint on lead curves that they cannot cross each other. The difficulty of lead curve detection lies in two challenges.

Unknown end-point of lead curve: the end-point of a lead curve is entirely unknown beforehand. A lead curve may end outside the object boundary, or on the object boundary, or at a certain position inside the object boundary.

Complex cross-points: a lead curve will cross many other curves that are misleading before it reaches the end. These cross-points can be the outer bound of the object, inner curves of the object and curves in the shaded area. Therefore, cross-points have different contexts.

Putting these two challenges together, complex cross-points are misleading for the detection process, especially when the end-point of the curve is unknown.

Previous curve detection algorithms [3–7] are usually applied in simple or no cross-point situation. In work [4], authors presented a novel method to detect curves with unknown end-points using minimal path techniques. Their method is evaluated on crack images, in which the cross-point situation is very simple. Authors in work [8] proposed a new and fast method for dominant point detection and polygonal representation of a discrete curve. They do not consider any cross-point situation.

We formulate the lead curve detection task as a problem of finding the best curve among all possible curves given the start-point. We make the following four contributions to tackle the challenges of unknown end-point of lead curve and complex cross-points.

- We introduce the "cross-point graph" representation which captures the topology of cross-point connectedness. We introduce a coupling concept "curve-path", which is a pair of the curve in the original image and the corresponding path in cross-point graph. Based on the cross-point graph representation and curve-path concept, we transform the lead curve detection task to a problem of searching in the cross-point graph for the path whose corresponding curve is most likely to be the lead curve.
- We design a set of joint feature representations for geometric plus context properties of a curve-path. The joint feature representations include smoothness at points, position relationship between the end node and object boundary, direction difference between the curve and the shaded area, angle difference between the curve parts inside and outside the object boundary and the length difference between the lead curve and other curves in the same image.
- We define a task specific loss function to measure the difference between two curve-paths. We also define a scoring function which gives higher score for the path of a groundtruth lead curve. We exploit max margin structured learning framework to learn the weightings of different joint feature representations. Furthermore, we design a mixed negative instance sampling strategy for the training stage and prune the search space significantly in the inference stage.
- We collect precise groundtruth by building a software to facilitate manual labeling and make the dataset publicly available.

The remaining of the paper is organized as follows. Section 2 introduces related work. Section 3 introduces the cross-point graph representation, describes the curve-path concept and transforms the lead curve detection task to a path searching problem on the cross-point graph. Section 4 gives the solution, which includes patent image parsing, joint feature representation design, inference and training. Section 5 reports extensive experimental results. Section 6 discusses the applications of lead curve detection in multimedia analysis. Section 7 draws some conclusions.

2. Related work

There have been many researches on contour detection in medical image analysis. A promising and vigorously researched technique is deformable model [9–13]. Priori knowledge about the location, size and shape can be embedded into the model. It has been widely applied in segmenting, matching and tracking anatomic structures. But the performance of deformable approach is usually affected by the initial condition and complex cross-points. In the case of patent image, when a lead curve goes into the internal of the object, there are many cross-points. Such situation is difficult for a deformable model to handle. The general particle filter (PF) framework is a powerful framework that has been applied in tracking and matching [14,15]. In work [15], authors relax the assumption of having ordered observations and extend the particle filter framework to estimate the posterior density by exploring different orders of observations. Another related field is contour detection, grouping and matching [14,16,17]. The general approach is to decompose a closed contour of a given model shape into a group of contour segments and match the resulting contour segments to edge segments in a given image.

In our work, we focus on detecting a particular kind of contour: lead curve. Generally speaking, our work solves the problem of contour detection in complex contour cross-point situation. Different from contour grouping in which edge broken is the major challenge, in our task cross-points are the major challenge. Cross-points will mislead the detection process. Our task is also different from contour matching where a template is given. It is difficult to give a lead curve template since both its length and cross-point status change dramatically among different images.

3. Problem formulation

In this section, we assume that the start-point is given and the image is a binary image where 1 s correspond to the drawing and 0 s correspond to the background. The details of extracting start-point and converting an image to a binary image will be described in Section 4.1. We formulate the problem by simulating the process that a human detects a lead curve.