Feature-related Searching Control Model for Curve Detection

Mingyi Zhang, Xilong Liu, De Xu, *Senior Member, IEEE*, and Zhiqiang Cao, *Senior Member, IEEE*

***Abstract*—In this paper, a novel method is proposed for curve detection in images using a feature-related searching control model. It is composed of three parts: prediction, searching, and updating. Firstly, curve related features are modeled to a three order array. Then, equations of the prediction, searching and curve parameter updating are deduced. Thirdly, an optimal model for curve parameter estimation during iterations is given. Based on the proposed model, a curve detection algorithm is designed. Experiments on thousands of images demonstrate the effectiveness and advantages of the proposed method. Comparison experiments with state-of-the-art methods show that the proposed method outperforms the existing methods on most indexes. Our method can describe the contents of original images more completely with fewer curves. The contour evaluation framework and the Berkeley Segmentation Dataset (BSDS) are used to evaluate the performances of different curve detection methods. The proposed method can also detect curves in the order relates to their importance, which has been validated in experiments.**

***Index Terms*— feature description, searching control model, prediction, searching, updating, curve detection.**

I. Introduction

CURVE detection methods have been widely used in many applications, such as image segmentation [1], image compression [2], pose estimation [3] etc. Due to extraordinary irregularity and randomness, how to extract complete curves reliably in different kinds of images becomes a very challenging task and is a long-standing problem in computer vision.

There are two main streams of curve detection methods, namely, the global fitting based and local based methods. Global fitting based methods utilize prior knowledge to fit simple curves such as straight lines, arcs and ellipses. One of the most famous global fitting method is Hough transform (HT) [4]. HT has a prominent advantage of robust detection results and is widely used in many applications. But there are also some deficiencies, such as ignoring correlations between edge points, which makes a large number of false detections. In addition, exhaustive voting mechanism pays for high computation. Therefore, many scholars make a wide range of explorations and improvements on the HT. In order to reduce the computation amount of HT, Xu *et al*. proposed Randomized Hough Transform (RHT) [5], in which the edge points are chosen randomly for voting rather than exhaustively. Zhang *et al*. incorporated gradient orientation information into RHT. To some certain extent, this method avoids invalid sampling and calculation [6]. Galambos *et al*. presented the Progressive Probabilistic Hough Transform (PPHT) [7] which has greatly improved the real-time performance of HT, and is often used for critically real-time applications. According to principles of probability, PPHT does sampling and fitting alternately, which means the more support points on one line segment has the greater probability it would be detected. In spite of all these improvements, the false connection is still not solved satisfactorily. The dimensions of the voting space is equal to the number the curve's analytical parameters. In applications, HT based methods can only be used to detect simple curves whose parameters is fewer.

Compared with global fitting based methods, local based methods can detect arbitrary curves without prior knowledge. One branch of these methods is growing algorithms which can be traced back to Freeman chain code [8]. Many researchers have made improvements based on Freeman chain code. Suzuki *et al*. determined topological relationships of borders in binary image, including the outer boundaries, the boundaries of holes and their hierarchies [9-10]. Chen *et al*. introduced constraints of the shortest path and the minimum orientation chaos to remove burrs and confused connections in thinned edge map, in order to extract main curves from messy edge maps [11]. Aiming at solving the multimodal image registration problem, Li *et al*. used a modified Canny operator to detect edges, then curves were extracted based on edges, finally line segments were fitted using the extracted curves [12].

Another branch is based on the Feature Support Region (FSR). FSR refers to a connected image region in which one or more features are consistent. Burns *et al.* first used directions of gradients as information for line segments detection, and put forward the concept of line support region [13]. Line support region refers to the rectangular areas where the inner pixels have similar gradient directions. The long axis of a line support region is approximately vertical to the inner gradient direction. Burns's method has a low false detection rate and line segments detected are more accurate. However, long line segments are often broken into small pieces due to interferences such as noises and clusters. To solve these problems, a large number of parameters need to be adjusted manually according to different cases which largely impeded the versatility of the algorithm. In addition, exhaustive search mechanism and gradient operation make it impossible to execute in real time. Philip *et al*. improved Burns's method by introducing a partitioning orientation mechanism into subareas, which accelerates the algorithm's speed partly [14]. Further, Desolneux *et al*. introduced

This work was supported in part by the National Natural Science Foundation of China under Grant 61503376, 61633020, 61733004, 61421004, and the Beijing Natural Science Foundation under Grant 4161002.

The authors are with the Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the University of Chinese Academy of Sciences, Beijing 101408, China. (*Corresponding author: Xilong Liu*, e-mail: xilong.liu@ia.ac.cn.)

the Gestalt cognitive theory and the Helmholtz principle to solve complex tuning problems in Burns's method [15]. The Helmholtz principle states that when the probability of a desired geometry occurring in nature noise images is less than a certain threshold, it can be considered that the geometry has the cognitive significance [16-17]. Desolneux proposed a line segment discrimination method to screen a large number of meaningless line segments, which largely preserves meaningful geometries of the original images. The robust screening mechanism that enhances the accuracy and significance of line segments makes the method much more versatile. Multisegment detection [18] is an important improvement over Desolneux's method. Grompone *et al.* solved the problem of multiplex detection in fuzzy boundaries according to the exclusion principle, and added weighting coefficients associated with collinear lines to ease the improper connection problem in Desolneux's method.

Grompone *et al*. improved a new search mechanism of line support region, and designed a fast line segments detector (LSD) [19], which is one of the most outstanding line segment detection algorithms by now. LSD improves exhaustive search strategy by sorting feature points into different levels according to gradient value, feature points in high level have the preference to be checked earlier. Moreover, the used points are marked as invalid points to avoid repeat searches. The efficiency of the algorithm is improved. The gradient orientation is used to construct FSR in most of the reported methods. Viorica improved LSD to ELSD by adding elliptical constraint rules to associate those line support regions with the potential to form an elliptic arc. ELSD uses different criteria to judge the membership of different geometries of the linked areas, and selects the most appropriate geometry (such as lines, elliptic arcs, etc.) [20]. It avoids the drawback of using large numbers of small line segments to describe elliptic arcs. However, it cannot resolve the problem of arbitrary curve detection. Besides, adaptability of ELSD is not ideal, which may mainly be caused by the preset rules. Guo *et al.* [34] proposed a multi-stage method for curve detection. The curve fragment search space is iteratively reduced by removing unlikely candidates using geometric constrains, which can avoid the drawback of using a global function directly on edges. Kolomenkin *et al*. [35] proposed a multi-scale curve detection method on surfaces. This method can automatically detect the optimal scale for each point on the surface. It is better than other methods because manual user intervention is not required. In addition, it does not require manual parameter tuning. Liu *et al*. [21] proposed a line segment detection method based on bionics, called Line Segment Perceptron (LSP), which is quite different from the traditional methods. Its main framework is a three-layer network inspired by the structure of visual cortex. The processing units in the network include bionic simple cells, complex cells and super complex cells.

The existing methods have tried to solve the problem of curve detection from different aspects. However, there are still some bottlenecks, which can be summarized as:

- Curves are detected only from the distributions of edge points or gradients, but cannot be detected from other contents with different features;
- The incompleteness of curves are more obvious when describing some contents such as the contour of an object, boundaries of image areas;
- Complex contents cannot be described with curves concisely, namely, descriptions with fewer curves.

It is difficult to solve above problems. Interestingly, a ground truth is that for human, curve detection seems to be a very simple task even when the observers are not familiar to the content. This implies that curve detection task is not simply a data-driven problem. It is achievable to solve the above problems to a great extend just by theoretical analysis which has been successfully used in many image processing tasks [22-23]. Generally speaking, to solve these problems, the first step is to generalize line segment detection to curve detection. More features should be used except gradients and edges. Detection should not only rely on local features. The second step is to improve the completeness of curves, which is one of the most challenging parts due to the absence of prior knowledge. Prediction is a possible way to break the limitation of local features, which is reflected in the prediction part of the proposed model. This is the core idea of this work. Besides, aiming at curve detection tasks, features are modeled into a uniform data format for deducing and analyzing.

The rest of the paper is organized as follows: In Section II, we present a uniform format to model curve related features. The curve detection method based searching control model including prediction, searching and updating is presented. Based on results deduced in Section II, a curve detection algorithm is designed in Section III. Results of both qualitative and quantitative experiments are presented in Section IV. Finally, the paper is concluded in Section V.

II. Curve Detection With Searching Control Model

The Kalman filtering method consisting of prediction, observation and updating can be properly imitated to realize curve detection in images. The curve detection method in this work includes three parts such as prediction, searching and updating.

First, it is necessary to set a curve constraint in order to build an easy-to-analysis model. A curve is composed of a series of continuously smooth curve primitives. In this paper, a strict definition of a smooth curve primitive is given as follows. The smooth curve primitive is a curve arc that can be approximated by a quadratic in (1).

$$au^2 + bv^2 + cu + dv + e = 0 \tag{1}$$

where u, v are the corresponding image coordinates. a, b, c, d and e are coefficients.

A. The Feature Array

Ignoring the specific types of features, the tangential related information at position (u, v) is described as a series of feature values, which can be obtained via different extractors in different scales and orientations. These values can be organized into a third-order feature array which implicit structural information. Because it is similar to a third-order tensor, some symbols and definitions from tensor analysis are introduced for convenience. The third-order feature array is denoted as $\mathbf{T}_{ijk}$.

whose components are outputs of feature extractor i in scale j with different orientation k, $i = 1, 2, \ldots, N_f$, $j = 1, 2, \ldots, N_s$, $k = 1, 2, \ldots, N_a$. The orientation k represents the direction with the angle $2\pi k/N_a$ rad.

The curve detection process is a continuous iterative process of prediction, searching and updating. In the iteration of the m-*th* step, the obtained feature array is denoted as $\mathbf{T}_{ijk}(m)$.

$$\mathbf{T}_{ijk}(m) = \mathbf{F}_{ijk}(u_m, v_m) \tag{2}$$

where (u_m, v_m) is the position of curve detection, $\mathbf{T}_{ijk}(m)$ is the third-order feature array, $\mathbf{F}_{ijk}(m)$ is the feature extraction function in the *m-th* step.

The feature array $\mathbf{T}_{ijk}(m)$ is used to get the curve related tangential information including tangent scale S_f, orientation θ_f and step length l_f .

$$\mathbf{f}_m = \Psi\left[\mathbf{T}_{ijk}(m)\right] \tag{3}$$

where $\mathbf{f}_m=[S_{fm}, \theta_{fm}, l_{fm}]^{\mathrm{T}}$ is a state vector related to feature array $\mathbf{T}_{ijk}(m)$. S_{fm} stands for the significant scale. θ_{fm} is the significant orientation and l_{fm} is the step length for searching. Ψ denotes the mapping from the feature array $\mathbf{T}_{ijk}(m)$ to the state vector $\mathbf{f}_m$.

The specific form of $\Psi(\mathbf{T}_{ijk})$ can be a manually designed analytical expression or an implicit expression learned from machine learning. One possible analytical expression is presented in the experiment part. More details are available in Section III.A, see (26) and (27).

B. Curve Position Prediction

The transfer position, significant scale, tangential direction and step length in the *m-th* step are expressed as a state vector ζ_m :$[S_m, \theta_m, l_m]^{\mathrm{T}}$. The state transition vector $\Delta\zeta_m$ from the (*m*-1)-*th* step to m-*th* step is defined as

$$\Delta\zeta_m = \zeta_m - \zeta_{m-1} \tag{4}$$

The state transfer vector of the (*m*+1)-*th* step is predicted as $\Delta\zeta^e_{m+1}$. In the absence of prior knowledge, prediction of the (*m*+1)-*th* step is based on both the feature array and history information of the *m-th* step, which is expressed in (5).

$$\Delta\zeta^e_{m+1} = \Phi\left(\mathbf{f}_m, \Delta\zeta_m\right) \tag{5}$$

where $\Delta\zeta^e_m$ is an estimated state transition vector $[\Delta S^e_m, \Delta\theta^e_m, \Delta l^e_m]^{\mathrm{T}}$. Φ is the prediction function, namely, the prediction part of the searching control model is designed as an uncoupled-weighted form:

$$\Phi\left(\mathbf{f}_m, \Delta\zeta_m\right) = \eta_m \odot \phi_f\left(\mathbf{f}_m\right) + \left(1-\eta_m\right) \odot \phi_p\left(\Delta\zeta_m\right) \tag{6}$$

where η_m is a weight vector with the same dimensions of $\Delta\zeta_m$. $\phi_f(\mathbf{f}_m)$ is a feature-based prediction function, it is current term; $\phi_p(\Delta\zeta_m)$ is a prediction function related to history variation tendencies of the curve segment, it is historical item.

If l_{m+1} is sufficiently small, the curve between (u_m, v_m) and (u_{m+1}, v_{m+1}) can be approximated as a straight line segment. This assumption can be satisfied in actual images since curves can be represented with very short line segments. In this case, $\phi_f(\mathbf{f}_m)$ can be obtained in an analytical form according to the geometric relationship as shown in (7).

$$\begin{bmatrix} \Delta u^{ef}_{m+1} \\ \Delta v^{ef}_{m+1} \\ \Delta S^{ef}_{m+1} \\ \Delta\theta^{ef}_{m+1} \\ \Delta l^{ef}_{m+1} \end{bmatrix} = \begin{bmatrix} l_{fm} & 0 & 0 & 0 & 0 \\ 0 & l_{fm} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_{fm} \\ \sin\theta_{fm} \\ S_{fm} \\ \theta_{fm} \\ l_{fm} \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ S_m \\ \theta_m \\ l_m \end{bmatrix} \tag{7}$$

In order to obtain the analytic expression of $\phi_p(\Delta\zeta_m)$, the curve to be detected is expressed in the form of parametric equations as in (8).

$$\begin{cases} u = g(t) \\ v = h(t) \end{cases} \tag{8}$$

where t is a self-variable, g and h are the functions of t.

The arc length of the curve from the initial position where t = 0 to the current position is shown in (9):

$$\hat{l} = \int_0^t \sqrt{\dot{g}^2 + \dot{h}^2}\,\mathrm{d}t \tag{9}$$

If the parameter t in (8) is defined as the length of the curve arc, (10) can be gotten by taking the derivative of t on both side of (9).

$$\begin{cases} \dot{g}^2 + \dot{h}^2 = 1 \\ \dot{g} = \dfrac{\mathrm{d}u}{\mathrm{d}\hat{l}} \\ \dot{h} = \dfrac{\mathrm{d}v}{\mathrm{d}\hat{l}} \end{cases} \tag{10}$$

According to the curve constraint in (1), we have:

$$2au\dot{g} + 2bv\dot{h} + c\dot{g} + d\dot{h} = 0 \tag{11}$$

The simultaneous solution of (10) and (11) is shown in (12).

$$\begin{cases} \mathrm{d}u = \pm\sqrt{\dfrac{(2bv+d)^2}{(2bv+d)^2 + (2au+c)^2}} \cdot \mathrm{d}\hat{l} \\ \mathrm{d}v = \mp\sqrt{\dfrac{(2au+c)^2}{(2bv+d)^2 + (2au+c)^2}} \cdot \mathrm{d}\hat{l} \end{cases} \tag{12}$$

Transition step length Δl_{m+1} is apparently in inverse proportion to the curvature of the curve, according to the local expression of curvature formula, the continuous analytic function can be written in the form of (13),

$$\mathrm{d}\hat{l} = \frac{K}{\left|\dot{g}\ddot{h} - \ddot{g}\dot{h}\right|} \tag{13}$$

where K is a constant.

The variation of the significant scale ΔS_{m+1} has no analytical relationship with the curvilinear parametric equation, and is entirely dependent on the variation of image content. Along the curve trend, image features are assumed to change smoothly, which can be predicted under the assumption that the second order differential is 0, as shown in (14).

$$\Delta S_{m+1} = 2\Delta S_m - \Delta S_{m-1} \tag{14}$$

Thus the analytic expression of $\phi_p(\Delta\zeta_m)$ can be obtained:

$$\begin{cases} \Delta u_{m+1}^{eh} = \pm \sqrt{\dfrac{\left(2b_m v_m + d_m\right)^2}{\left(2b_m v_m + d_m\right)^2 + \left(2a_m u_m + c_m\right)^2}} \cdot \Delta l_m \\ \Delta v_{m+1}^{eh} = \mp \sqrt{\dfrac{\left(2a_m u_m + c_m\right)^2}{\left(2b_m v_m + d_m\right)^2 + \left(2a_m u_m + c_m\right)^2}} \cdot \Delta l_m \\ \Delta S_{m+1}^{eh} = 2\Delta S_m - \Delta S_{m-1} \\ \Delta\theta_{m+1}^{eh} = \arctan(\Delta v_{m+1}^{eh} / \Delta u_{m+1}^{eh}) - \arctan(\Delta v_m / \Delta u_m) \\ \Delta l_{m+1}^{eh} = \dfrac{K}{\left|\dot{g}\ddot{h} - \ddot{g}\dot{h}\right|} - l_m \end{cases} \quad (15)$$

where $\dot{g}$ and $\ddot{g}$ can be approximated by (16).

$$\begin{cases} \dot{g} = \dfrac{\Delta u_m}{\Delta l_m} \\ \ddot{g} = (\dfrac{\Delta u_m}{\Delta l_m} - \dfrac{\Delta u_{m-1}}{\Delta l_{m-1}}) / \Delta l_m \end{cases} \quad (16)$$

Similarly, $\dot{h}$ and $\ddot{h}$ can be obtained.

C. Curve Point Searching

In a trajectory measuring task, which is based on the Kalman filtering, predictions are modified with observations to get the optimal estimations. For a curve detection task, feature arrays are taken as observations. Estimations of the observation can be simplified as finding the image position whose feature array is "the best". Once the form of the "the best" feature array is determined, the image position search task becomes a feature matching problem. The relationship of feature arrays of the current curve point and the next potential curve point can be modeled as a function about the position and scale of the next potential curve point. If the distance between the current curve point and the next potential curve point is small, the expression of the function can be well approximated by a numerical approximation method. An optimal function is proposed to estimate the feature array. Taking the last step as the m-*th* step, the optimal function is expressed as:

$$\mathbf{f}_{opt} = \operatorname{argmin} \left\| \mathbf{T}_{ijk}(u,v) - \Psi^{-1}(\mathbf{f}_{m+1}) \right\| \quad (17)$$

where $\Psi^{-1}(\mathbf{f}_{m+1})$ is the inverse function of (3). $\mathbf{f}_{m+1}$ is the feature related state vector $[S_{fm+1}, \theta_{fm+1}, l_{fm+1}]^{\mathrm{T}}$ of the potential curve point.

There are three independent variables S_{fm+1}, θ_{fm+1} and l_{fm+1} need to be solved. It should be noted that, l_{fm+1} is set as l^e_{m+1} directly, which can be gotten by adding l_m and Δl^e_m. (17) can be solved with a search policy. Based on the assumption that the tangential direction of a curve changes smoothly, it can be known that errors between $\Delta\zeta^e_{m+1}$ and $\Delta\zeta_{m+1}$ will be acceptable. For l_{m+1} is constant during the optimization, the searching area can be locked on a short arc which takes (u_m, v_m) as the center point and l_{m+1} as its radius. The feature array on the arc that satisfies (17) is the optimal estimation of the actual feature array. Then, at the $(m+1)$-*th* step, $\Psi^{-1}(\mathbf{f}_{m+1})$ can be taken as a transformation of the feature array at the m-*th* step according to $\mathbf{f}_m$, which can be modeled as a feature-transfer function expressed as:

$$\Psi^{-1}(\mathbf{f}_{m+1}) = \Gamma[\mathbf{T}_{ijk}(m), \mathbf{f}_m] \quad (18)$$

It is known that the feature array $\mathbf{T}_{ijk}$ only records features at a series of discrete directions and scales, exact values of θ_m and S_m from the state vector $\mathbf{f}_m$ are unknown. The feature response surface is denoted as $f_i{}^m(\theta, S)$. With other values recorded in the feature array, a surface defined on the θ-S plane can be fitted, which can be expressed as:

$$f_i{}^m(\theta, S) = \mathrm{Fit}[\mathbf{t}_{i,j',k'}(m)] / f_{i,}(\theta_m, S_m) \quad (19)$$

where $\mathbf{t}_{i,j,k'}$ represents an sub-array of $\mathbf{T}_{ijk}$ with constant k'.

$f_{i,}{}^{m+1}(\theta, S)$ can be approximated as a translation in (20).

$$f_i{}^{m+1}(\theta, S) = f_i{}^m(\theta - \theta_{m+1} + \theta_m, S - S_{m+1} + S_m) \quad (20)$$

Then θ_{fm+1} and S_{fm+1} can be obtained from $f_{i,}{}^{m+1}(\theta, S)$ via searching the local maximum. Thus $\mathbf{f}_{m+1}=[S_{fm+1}, \theta_{fm+1}, l_{fm+1}]^{\mathrm{T}}$ is determined. Then $\Psi^{-1}(\mathbf{f}_{m+1})$ can be gotten. Submitting $\Psi^{-1}(\mathbf{f}_{m+1})$ to (17), the optimal solution (u, v) is taken as a point on the curve.

D. Curve Parameters Updating

The curve parameters *a*, *b*, *c*, *d* and *e* are updated in each step. Taking these parameters as variables, optimal values of a_m, b_m, c_m, d_m and e_m at the *m-th* step are the solution of (21).

$$\min \sum_{i=1}^{m} f_i(a_m, b_m, c_m, d_m, e_m)^2 \quad (21)$$

where $f_i(a_m, b_m, c_m, d_m, e_m) = a_m u_i^2 + b_m v_i^2 + c_m u_i + d_m v_i + e_m$.

A reasonable updating procedure can be designed with a model in the parameter space. The parameter space of curves is denoted as $\mathbf{P}$. Parameters of the curves passing a given point (u_i, v_i) span a subspace of $\mathbf{P}$, denoted as $\mathbf{S}$. $\mathbf{S}$ is called the solution space of the *i*-th step, because it is composed of the solutions of $f_i(a,b,c,d,e)=0$. If the element $\mathbf{e}_m=(a_m, b_m, c_m, d_m, e_m)$ in $\mathbf{P}$ is not contained in $\mathbf{S}$, the distance from $\mathbf{e}_m$ to $\mathbf{P}$ is proportional to the error, namely, the value of $f_i(a_m, b_m, c_m, d_m, e_m)^2$, and an optimal path from $\mathbf{e}_m$ to $\mathbf{P}$ is along the direction of gradient descent, and at $\mathbf{e}_m$ the direction is calculated as (22):

$$\mathbf{G}r_m = -\nabla[f_i(a_m, b_m, c_m, d_m, e_m)^2] \quad (22)$$

where $\mathbf{G}r_m$ is the gradient direction vector.

Finding curves that pass a series of points is to determine a coincide space $\mathbf{C}$ of these points' solution spaces. It is sure that the solution spaces are convex. So, starting from $\mathbf{e}_m$ it will always reach $\mathbf{C}$ along the direction synthesized by the gradient descents calculated with each point.

The parameters can be approximated by adjusting the parameters in a small length along the gradient descent of the current point. The formula can be given as follows.

$$\begin{cases} \mathbf{G}r_m = -(\dfrac{\partial f_m{}^2}{\partial a}, \dfrac{\partial f_m{}^2}{\partial b}, \dfrac{\partial f_m{}^2}{\partial c}, \dfrac{\partial f_m{}^2}{\partial d}, \dfrac{\partial f_m{}^2}{\partial e}) \\ \left\|\Delta \mathbf{e}_\mathbf{m}\right\| = \left\|\mathbf{G}r_m\right\| \\ \mathbf{e}_{\mathbf{m}+1} = \mathbf{e}_\mathbf{m} + \varepsilon \left\|\Delta \mathbf{e}_\mathbf{m}\right\| \mathbf{G}r_m \end{cases} \quad (23)$$

where ε is the updating rate.

For polynomial curve functions, solution spaces are hyperplanes in $\mathbf{P}$, and they are always convex spaces apparently. So the convergence of (23) is guaranteed for ideal curves by which all the given points are passed. Noises will make a curve point be away from the right place, then the updating points cannot be passed by the same curve. In this condition, the solution spaces have no coincidence part. (23) will converge into a wrong place

that minimizes (21). Because random noises are isotropous, the convergence of (23) will be ensured when the update step length ε is small enough. Because the influence is hard to be analyzed quantitatively, experiments are performed and the results show the convergence of (23) is robust to strong noises.

III. Algorithm Implementation

A. Algorithm Description

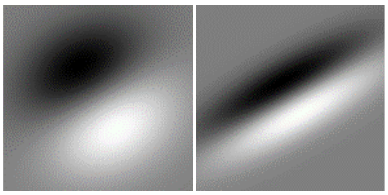
Feature extraction is very important for curve detection. There are many feature extraction methods. Bai *et al.* [31] proposed a VHR object detection method based on structural feature description and query expansion. A discriminative feature was presented to improve the performance of current bag-of-word based image classification process [32]. Liu *et al.* [33] proposed an object detection method via structural feature selection and part-based shape model with contour feature descriptor. In addition, Gabor-type convolution kernels have shown good performances for image descriptions in both bionics [24] and deep learning [25, 26]. They are used in this work for feature detection. The analytical expression is as follows.

$$f(\theta,\mu_u,\mu_v,b)=\alpha\mu_u^{-0.5}\mu_v^{-1}e^{\frac{(u\cdot\cos\theta-v\cdot\sin\theta)^2}{2\mu_u^2}+\frac{(u\cdot\sin\theta+v\cdot\cos\theta)^2+b^2}{\mu_v^2}} \quad (24)$$

where α is a constant, μ_u, μ_v and b are parameters of the convolution kernel, θ stands for the orientation.

In our algorithm, two different kinds of kernels, 6 orientations and 3 scales are used. The extracted feature array is $\mathbf{T}_{ijk}$. Fig. 1 shows images of two kernels generated with different parameters. Fig. 2 shows a feature map in a color image, where the large scale (j = 1), middle scale (j = 2) and small scale (j = 3) are demonstrated with R, G, B channels. For each channel, the maximum value from 12 feature values (2 types and 6 orientations) is chosen to show.

ALGORITHM 1 Curve Segment Detector
Input: a preset integer N. feature map F.
Output: a set of curves {Output set}.
Functions: seed points judge-function $\boldsymbol{f}(\mathbf{T}_{ijk})$;
seed point priority calculator $\boldsymbol{w}(P,\mathbf{T}_{ijk})$;
searching algorithm **Search**($P_{\text{start-point}}$, left or right);
curve linker **Interlink**(Branch_0,Branch_1).
Pre-processing: seed points map generation
For each point P in F
  if $f(\mathbf{T}_{ijk})$
    $P_{\text{priority}}=w(P,\mathbf{T}_{ijk})$;
    $\{P_{\text{seed-point}}\}\leftarrow(P,P_{\text{priority}})$;
  end
end
Searching: curves detection
for n< N
  $P_{\text{start-point}}\leftarrow\max\_P_{\text{priority}}\{P_{\text{seed-point}}\}$;
  Branch_0←**Search** ($P_{\text{start-point}}$,left);
  Branch_1←**Search** ($P_{\text{start-point}}$,right);
  Curve_segment←**Interlink** (Branch_0,Branch_1);
  {Output set}←Curve_segment;
end



(a) (b)

Fig. 1. Images of two kernels with different parameters. (a): θ=π/6, μ_u=0.5, μ_v =0.4, b =0.04; (b): θ=π/6, μ_u=0.7, μ_v =0.2, b =0.04

(a) Original image (b) The feature image

Fig. 2. The feature image of the proposed feature detectors

The proposed searching control model needs a start point. For curve detection task, the start points are called as seed points. In the designed algorithm, the features detected at point P is denoted as $\mathbf{T}_{ijk}(P)$. For any i, k, if (25) is satisfied then P is taken as a seed point.

$$\max\left[\mathbf{t}_{i',j,k}(P)\right]>\max\left\{\max\left[\mathbf{t}_{i?,j,k}(P_{adj})\right]\right\} \quad (25)$$

where P_{adj} is an adjacent point of P, and $\{\max[\mathbf{t}_{i',j,k}(P_{adj})]\}$ is a set of corresponding maximum values of all adjacent points. A search algorithm based on the searching control model as described in Section II is used to detect curves from seed points. Every seed point has a priority to be active. In our algorithm, the priories are generated according to the following rules:

(1) The larger the maximum feature value is, the higher priority the point has.

(2) If the salient feature is detected in a larger scale, the point should have higher priority.

(3) If the feature detected at a point is more regular, the point should have higher priority. This can be qualified by the difference between feature vectors detected in different scales.

Based on these rules, (26) is designed to calculate the priority of seed point SP quantitatively.

$$pri(SP)=\frac{(C_1-j^*)\max\left[\mathbf{t}_{i',j',k'}(SP)\right]}{C_2+\sum_{i,k}\mathrm{V}\left(\mathbf{t}_{i,j',k}\right)} \quad (26)$$

where $\mathrm{V}(\mathbf{t}_{i,j',k})$ represents the variance of all elements in array $\mathbf{t}_{i,j',k}$, C_1, C_2 are preset constants, j^* is the scale where the salient feature is detected. A seed point map generated with (26) is shown in the bottom left of Fig. 3.

In one iteration of the designed algorithm, curves are detected from the seed point whose priority is the highest. When the detection procedure is done, the seed point's priority will be set to null. The iteration process goes on until there are no effective seed points, the total curve number reaches the preset. ALGORITHM 1 gives more details.

When a state transition based searching algorithm starts, the state vector is initialized according to the feature detected at the seed point, denoted as ζ_0: $[S_0, \theta_0, l_0]^{\mathrm{T}}$, where (u_0, v_0) is the position of the seed point, and $[S_0, \theta_0, l_0]^{\mathrm{T}}$ is initialized as $[S_{fm}, \theta_{fm},$

(a) The original image

(b) The seed map of (a)

(c) Illustration of the curve detection process

Fig. 3. Details of the searching control model for curve detection

$l_{fm}]^{\mathrm{T}}$ using (3). An expression of (3) is designed in (31).Wherein i^* is the index of the salient features. $\mathrm{V}(\mathbf{t}_{i,j',k})$ is used to measure the chaos degree of image content.

$$\begin{cases} S_{fm} = S_{j^*} \\ \theta_{fm} = \theta_{k^*} \text{ or } \theta_{fm} = \theta_{k^*} + \pi \\ l_{fm} \propto 1 \Big/ \sum_{i,k} \mathrm{V}(\mathbf{t}_{i,j',k}) \\ j^* = \arg\max \sum \mathbf{t}_{i',j,k'} \\ i^* = \arg\max \sum \mathbf{t}_{i,j',k'} \\ k^* = \arg\max t_{i^*,j^*,k} \end{cases} \tag{27}$$

When features are chaotic, the corresponding curve may have a large curvature, and a small search step is reasonable, so the search step l_{fm} is designed to be in inverse proportion of $\mathrm{V}(\mathbf{t}_{i,j',k})$. Feature's orientation range is $[0, \pi)$. So, for each θ_{k^*}, there exist two opposite search directions, corresponds to the positive and negative conditions of du and dv in (12). In search algorithm, except the seed point, the alternative value of θ_{fm} can be determined under the constraint that between two adjacent iterations, the change of θ_{fm} is smaller than π. For every seed point, two searching procedures are started correspondingly, and the two detected curves are combined then. Details of the searching control model are demonstrated in Fig. 3. It can be seen that Fig.3(a) is the original image, Fig.3(b) is the corresponding seed map of Fig.3(a). In order to show the curve detection process of the searching control model more clearly, a seed point in the red circle of Fig. 3(b) is selected as an example, which corresponds to an red rectangular area in Fig.3(a). Fig.3(c) is the illustration of the curve detection process. During the *m-th* iteration of the searching, the point determined at the (*m*-1)-*th* step is selected as a starting point which is marked as a red square, and then the prediction point of the *m-th* step can be gotten, which is marked as a yellow square. Based on the prediction, the point's position is modified by observations. The pixels passed by the blue arc are observed, and the point meets (24) is considered to be the best estimation of the *m-th* iteration. The searching algorithm ends when the best estimation cannot be found or the image boundary is reached.

B. *Complexity of the Algorithm*

The computation complexity of this method mainly involves the feature detection part and the curve estimation. The computational complexity of the feature detection part is related to the number of pixels and the number of convolution operators. When the types and sizes of the convolution kernels are determined, the computational complexity is only related to the number of pixels. It is known that the computational complexity of a convolution operator is $O(N^2)$. If there are K convolution operators, then the computational complexity is $O(\mathrm{K}*N^2)$. Since the convolution operator used in this paper is finite, the approximation complexity of the feature extraction part is $O(N^2)$.

In the curve detection part, for any of the curve points, the calculation process is related to (7), (15), (17) and (24). Operations in (7), (15) and (24) can obviously be done in linear time. Although (17) is a search problem, the search area is limited in a very small range. Therefore, the calculation amount at each point is small and limited. During the curve detection, the worst case is that each point in the image is a curve point, and pixels in the image need to be traversed, in this case the calculation complexity can be taken as $O(N)$. In fact, curve points in the image are generally very sparse, and the computational complexity will not exceed $O(N)$.

Generally speaking, the computational complexity of our method is $O(N^2)$.

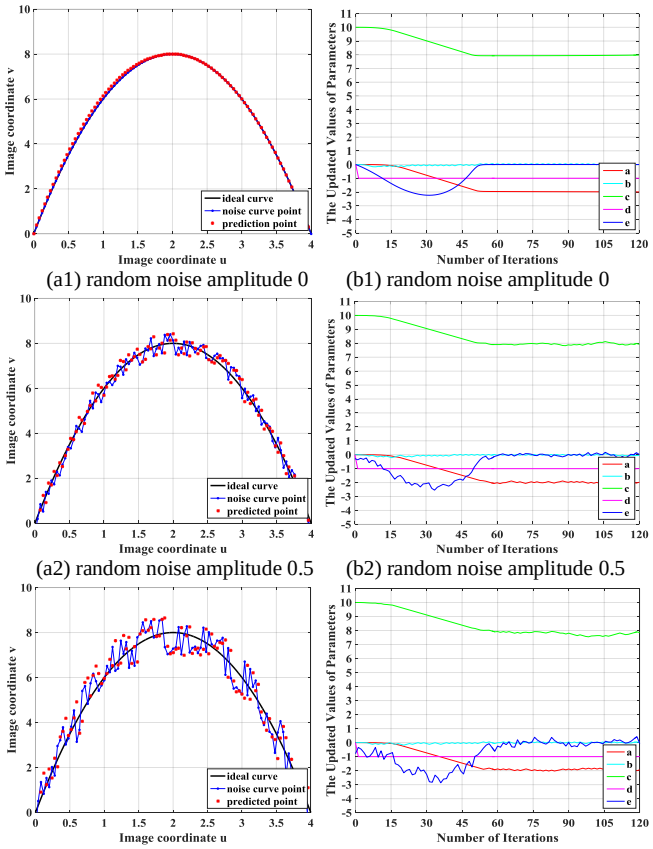
IV. Experiments

First, a large number of images are used to compare the effects of the proposed method and other methods. Then, the BSDS dataset is used in the comparison experiments to show performances of different methods.

A. *Experiments with Simulation Images*

The searching control model expressed in (15) combined with (20) is the theoretical basis of the proposed method. Before testing the whole algorithm with real images, quantitative validations of (15) are necessary. For this purpose, some sim-

ulations are designed. In a simulated image, a curve passing the original point with preset parameters is drawn. The original point is chosen as the seed point, and in (6), the feature term is set to 0, where η_m=0. During modification discussed in section II.D, the point (u_m, v_m) on the curve nearest to the predict point of the last iteration is submitted to (15). Ideal quadratic curves are tested firstly, then gradually stronger noises are added. Fig. 4 shows one example of these simulations. In image Fig.4(a1), (a2), and (a3), the generated curves are drawn in blue lines. Red asterisk "*" represents the predicted points according to (15). Images Fig.4(b1), (b1), and (b3) show the updated curves of curve parameters according to (20). The preset values of curve parameters are a=-2, b=0, c=8, d=-1, and e=0. Image Fig.4(a1) is the ideal curve without noises. Curves in image Fig.4(a2) and (a3) are added with random noises of different amplitude. Image Fig.4(b1), (b2), and (b3) are corresponding curves of updated parameters. It can be seen that parameters a, b, c, d and e converge to the true values quickly after several adjustments, even if there are strong noises. Through several simulations, it can be found that the parameters always converge to the true values. It seems that noise will not significantly affect the convergence rate, but only affect the prediction accuracy. The simulations confirm the correctness of the method's theoretical basis, and indicate (15) is reliable and effective.



(a1) random noise amplitude 0 (b1) random noise amplitude 0

(a2) random noise amplitude 0.5 (b2) random noise amplitude 0.5

(a3) random noise amplitude 1.0 (b3) random noise amplitude 1.0

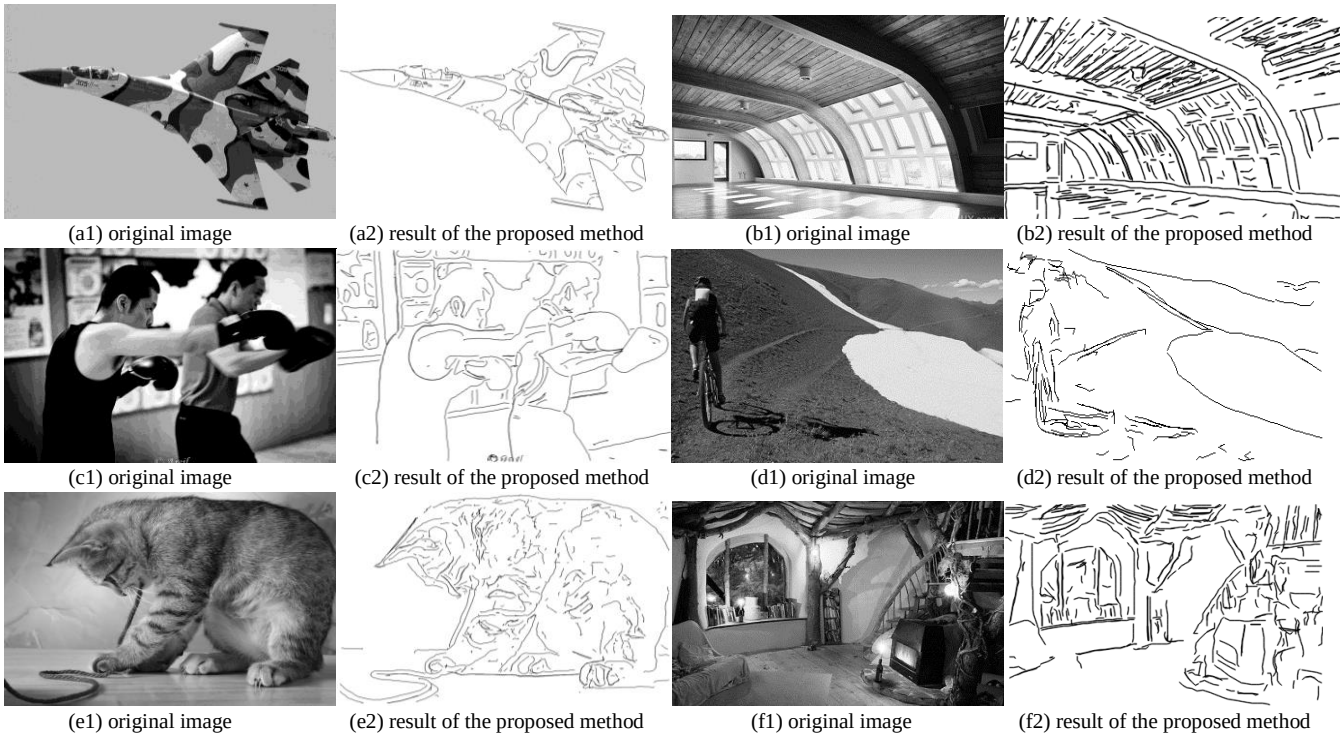
Fig. 4. Validation of the searching control model with simulated images

B. *Experiments with Actual Images*

Experimental results of the proposed method with several images are shown in Fig. 5. Before discussion, we should make an explanation that why output curves of different images seem to have different shades and width. In fact, this is only a visual illusion caused by scaling. Because the input images are in different size, different ratios are used to scale them in the proper size for show. The ratio is large when the image is large, and the curves will be scaled to a small width. Note that the detected curves represent well both the structures and details of these diverse images. Images Fig.5(a1) and (b1) contain highly geometrical contents with long curves in different shapes. Almost all the expected curves were found. Images Fig.5(c1) and (d1) contain some non-geometric objects such as human and snowfield. They are also well described by the detected curves precisely and concisely. Although most part of Fig.5(c1) is seriously defocused, the proposed method is robust enough to deal with it due to the prediction and updating mechanism which can extend the searching algorithm from clear seed points to blurred areas. Contents of images Fig.5(e1) and (f1) are highly complex and non-geometric. Details are almost completely detected with a relatively small amount of curves. The hairy back of the cat in image Fig.5(e1) is remarkable. It is a typical content which can be modeled by noise curves shown in Fig. 5. The cat's back is successfully detected as a long curve that further validates the correctness and effectiveness of the proposed searching control model.

A prominent advantage of the proposed method is the number of curves to be detected can be tuned. So far as we know, only HT-based algorithms have a similar function, in which the output curve numbers can be set manually. However, the tunable parameter is only the output number, but not the detect number, which means it is irrelevant to the algorithm's computation amount. Besides, in HT-based algorithms, which curves to be outputted are only decided according to the weight in voting space. This makes the output order of the detected curves is aimless. A distinctive property of the proposed method is that the curves are detected in the order relates to their importance. The longer, clearer and more salient curves tend to be detected earlier. As shown in Fig. 6, the curve detection process are similar to those done by a human artist. This allows the method to be used for foreground segmentation, image structure analysis and focus area detection. In image Fig.7(a1), there is an irregular object in the foreground and it is very difficult to segment precisely from the complex background. When the curve number is set to smaller than 30, the contours of the object in the foreground are precisely detected while the contents in the background are effectively ignored. Image Fig.7(b2) and (b3) show the process of an aerial image. With a small curve number, structure curves of the building are selectively detected, which can be used for some structure analysis tasks such as vanishing point detection and UAV pose estimation. In image Fig.7(c1), one part is in focus while the others are defocused. When the curve number is small, curves belonging to the focused area will be detected firstly. If necessary, with large curve numbers, details in whole image can also be successfully detected.

C. *The Comparative Experiments*

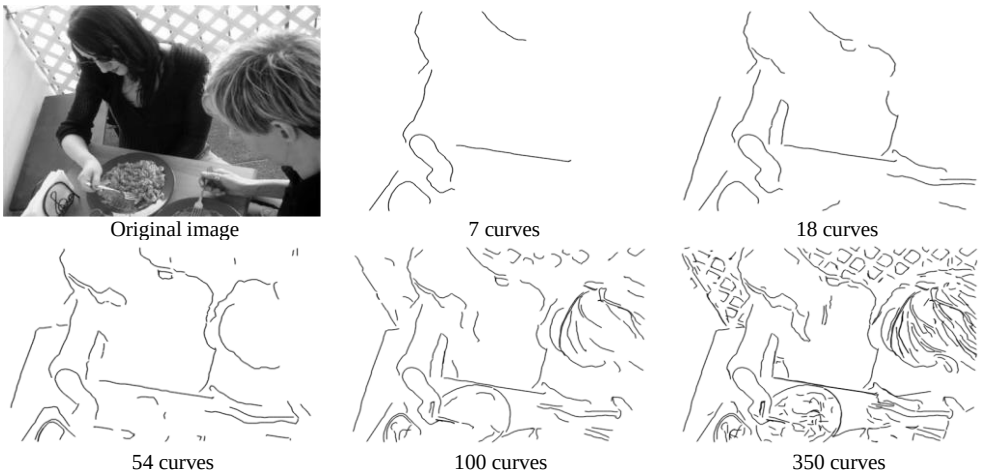


(a1) original image (a2) result of the proposed method (b1) original image (b2) result of the proposed method

(c1) original image (c2) result of the proposed method (d1) original image (d2) result of the proposed method

(e1) original image (e2) result of the proposed method (f1) original image (f2) result of the proposed method

Fig.5. Validation of the searching control model with actual images



Original image 7 curves 18 curves

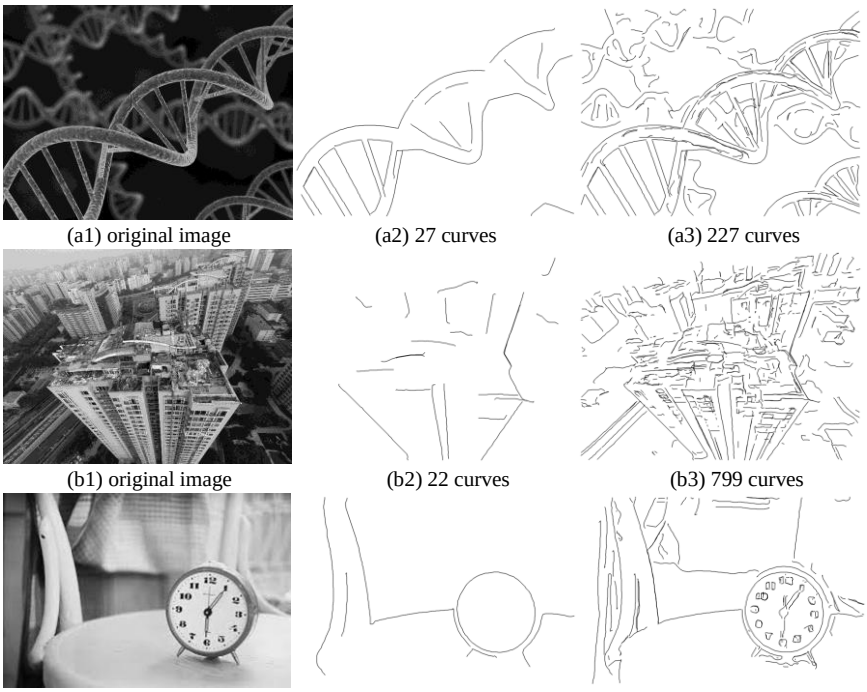
54 curves 100 curves 350 curves

Fig. 6. Results with gradually increasing curve numbers.

Due to the lack of a specific benchmark data set for curve detection algorithms, the proposed method is tested on thousands of images with different kinds, origins, and sizes. It should be emphasized that all of the tests and comparison experiments are done without tuning any parameter at all. Numbers of the output curves are set to unlimited and the algorithm ends when there are no available seed points and output all curves longer than a preset value.

Fig. 8 shows the comparison results with the state-of-the-art methods, including LSD, ELSD, and LSP. Some HT-based methods are also well developed and widely used. However, they will fail without prior knowledge about the curves. When using many small line segments to describe long curves, HT-based methods perform much worse than the above three methods. So, they are not compared here. Evaluation indexes of curve detection methods include number of curves detected, completeness of the original content described by the curves and the continuity of the curves detected from a continuous contour. An ideal method can describe the original contents completely with fewest curves, and the curves never break at continuous contours. In order to make quantitative evaluations, curves numbers of each methods are presented.

For image Fig. 8(a1), LSD and LSP describe the long curves as a series of short line segments and the numbers of line segments outputted by them are 134 and 230, separately. Different from them outputting straight line segments only, there are arcs outputted by ELSD for curve description and this makes the results smoother. The number of output curves is reduced to 111. However, arcs in ELSD are not long enough. Desired long curves are often broken into many small arcs and line segments with obvious gaps between them. The proposed method outputs only 4 curves and they reflect the image content



(a1) original image (a2) 27 curves (a3) 227 curves

(b1) original image (b2) 22 curves (b3) 799 curves

(c1) original image (c2) 7 curves (c3) 148 curves

Fig. 7. Potentials for some difficult problems.

smoother and more accurately than the existing methods without undesired gaps. For image Fig.8(b1), the results of LSD and ELSD lose some important information. For example, details of the Tom's face are not kept. Outputs of LSP look more complete. However, too many line segments are outputted with some undesired overlaps which make the result look trivial. Compared with the existing methods, the proposed method performs the best. Firstly, the number of output curves is the fewest. Secondly, details are described completely, similar to LSP but better than LSD and ELSD. Thirdly, there are few false detections and undesired overlaps, similar to LSD and ELSD, but better than LSP. A holistic and sophisticated curve detection method should be able to deal with different kinds of nature images. In fact, comparing to the existing state-of-the-art methods, advantages of the proposed method seem to be more significant when dealing with complex nature images.

From image Fig.8(c1) to (f1), comparison results of some very challenging images are present. Advantages of the proposed method mainly represented in three aspects: the image contents are described more completely with fewer curves; contours are detected as longer curves with fewer undesired gaps; output curves are smoother and more precise. Image Fig. 8(c1) is a photo of a swimming pool whose bottom is seriously distorted by the water wave. There are 527 curves detected, fewer than 902 by LSD, 771 by ELSD and 1152 by LSP. Information loses seriously in the results of LSD and LSP. ELSD outperforms them in completeness but two big circles are false detected. The proposed method outperforms all of them in completeness with almost no false detections. Note that most of the highly irregular curves are detected continuously without undesired breaks. This is why the output curves are much fewer than the traditional methods. For image Fig.8(d1), contours and textures of the zebra are much better detected by the proposed method, which makes the result look more beautiful. The legs of the zebra in the grass is remarkable. Also, contours of three zebras in the background are also successfully detected. For image Fig.8(e1), the irregular shaped buildings are described concisely with much fewer gaps by the proposed method, and at the same time, details of the city are successfully detected as curves with different lengths. The results of image Fig.8(f1) clearly demonstrate the advantage of smooth and precise of the proposed method. The contours of the woman's face are detected as graceful long curves.

D. Evaluations on BSDS

Evaluations of curve detection methods from only human observations are lack of persuasion. In order to give more authoritative evaluations, a targeted evaluation methodology with a specific dataset is necessary. Although there are still no such evaluation systems, frameworks used to evaluate the contour detection can be taken into effort with some modifications. Contour detection algorithms output a set of separate points, while the curve detection algorithms output a set of curve functions corresponding to some point chains. When drawing their outputs in the form of gray images, they look very similar because the relationship of points in a chain cannot be directly shown. In the dataset used for evaluating contour detection algorithms, the ground truth given by human is indeed drawn in curves and then divided into separate pixels. So evaluating curve detection methods with these datasets are reasonable. A precision-recall framework proposed by [27] is adapted to evaluate the curve detection algorithms on BSDS [28]. In this experiment, each of the outputted curve function is approximated as a set of discrete curve points. The curve detection is formulated as a classification problem of discriminating non-curve from curve pixels and the famous precision-recall framework [29] using human-marked curve points from BSDS is applied. Precision stands for the fraction of the detected curve points which are also edge points rather than other curve points, while recall is the fraction of the edge points that are detected as

curve points rather than that are not detected. When each of the discrete curve point is modeled as a probability distribution, our evaluation measure—the precision- recall curve can be obtained using the method proposed in [29].

Fig. 9 shows the performance of the proposed method (aliased as ct in the legend) compared to LSD, ELSD and LSP. The maximum F-measure [30] defined as:

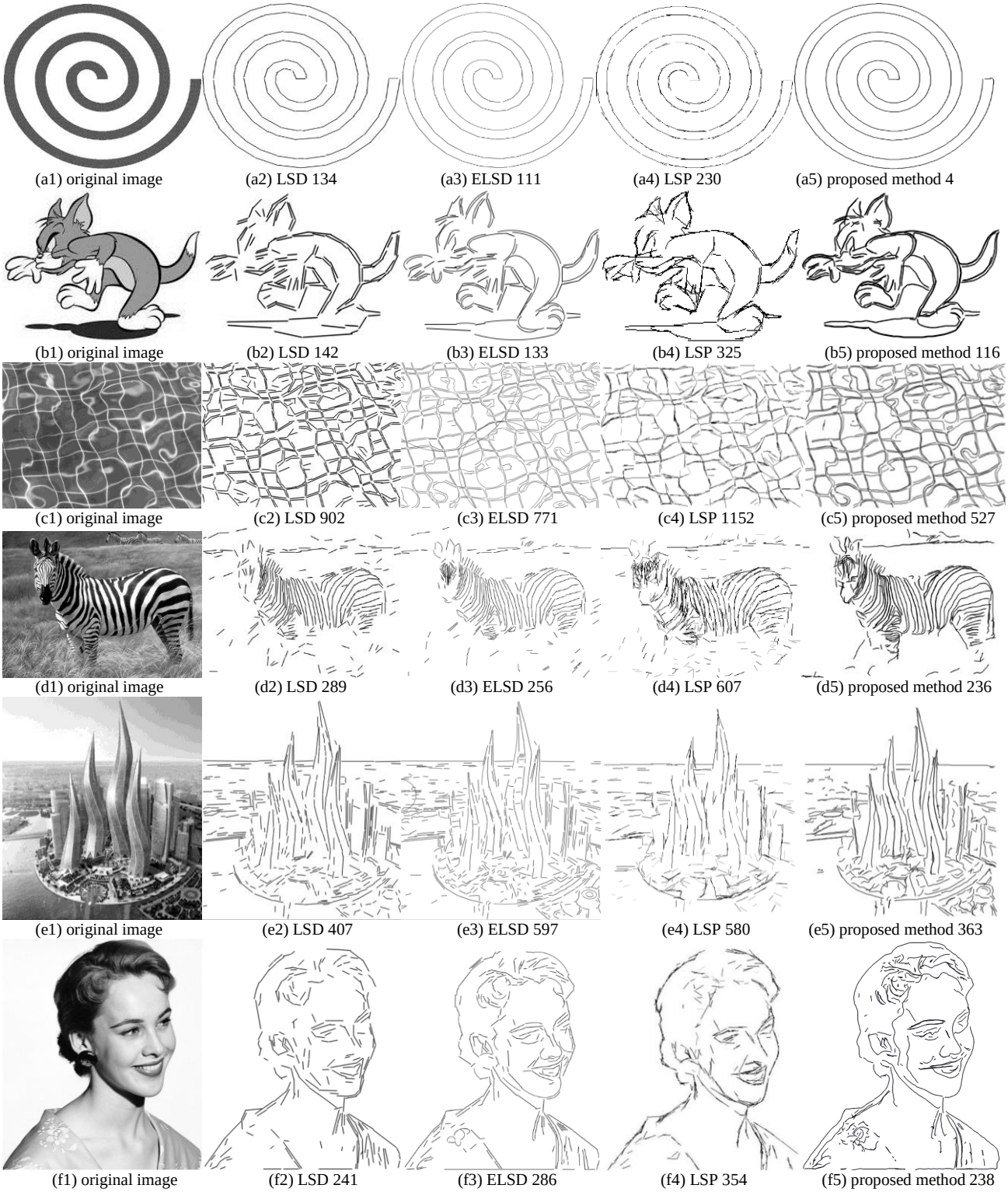


Fig. 8. Comparison with existing algorithms in images with typical curve contents. In each row, from left to right, the original image, results of LSD, results of ELSD, results of LSP and results of the proposed method. The number under each image is the corresponding curve number.

$$F = \frac{P \cdot R}{\alpha \cdot R + (1-\alpha) \cdot P} \tag{28}$$

where P is precision, R is recall, α is a factor. F captures the trade off as the weighted harmonic mean of P and R. When α is set to 0.5, the location of the maximum F-measure along each curve is shown in the legend. The highest maximum F-measure indicates that the proposed method has the best performance. This is a valuable quality for measurement applications.

$$F_m = \frac{P \cdot R}{\alpha \cdot R + (1-\alpha) \cdot P} \cdot \frac{C}{N} \tag{29}$$

Different from contour detection methods, performance of curve detection methods may be different even when their outputs have the same appearance. In this condition, the fewer the outputted curve number is, the better the performance is. Therefore, the framework for contour detection methods will be more reasonable when the curve number N is added as an evaluation parameter, as shown in (29). Correspondingly, F-measure is modified to F_m-measure by adding C and N, where C is a constant.

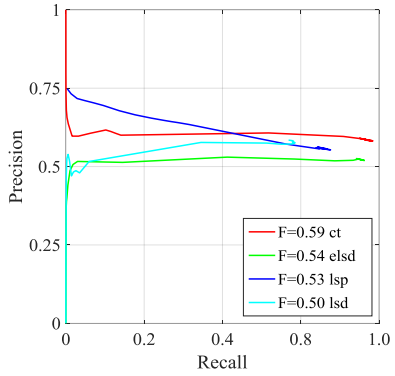


Fig. 9. The precision-recall curves of different curve detection methods

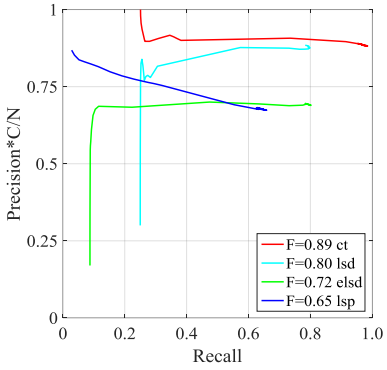


Fig. 10. The modified precision-recall curves of different curve detection methods testing on the BSDS dataset

As shown in Fig. 10, the advantages of our method are more obvious with the modified evaluation framework. The F_m-measure of our method is also the highest of all. The precision-recall curves show that the proposed method has relatively higher precision with strict recall, which means that our method can detect more potential curve points with fewer curves. The validity of our method is effectively illustrated.

V. Conclusion

Aiming at the curve detection task in images, we propose a searching control model in which prediction, searching and updating are done with both local features and history information. The deducing of the feature term is based on the proposed uniform format rather than other specific feature detectors, so that all anisotropic features will have the potential for curve detection tasks. The history information term is deduced with the assumption that the detected curve can be fit into a quadratic polynomial locally, which can meet the demand of precision in most applications. The algorithm is designed under the guidance of the proposed searching control model. Simulations with given curves confirm the correctness of the model. Testing experiments with thousands of actual images demonstrate its effectiveness. The completeness and conciseness of the detected curves can be intuitively seen from the result images. Qualitative comparison experiments with state-of-the-art methods show that the proposed method is superior to the existing methods on most visible indexes. It can describe the contents of original images more completely with fewer curves and tune curve numbers according to salience. Quantitative comparison with an authoritative evaluation is given using a contour detection framework and the database BSDS. The experimental results indicate the proposed method is superior to existing methods. Experiments also show the proposed method may potentially be used for some difficult problems such as foreground segmentation, image structure analysis and focus area detection.

References

[1] M. Wang, Q. Cui, and J. Wang, *et al*. "Raft cultivation area extraction from high resolution remote sensing imagery by fusing multi-scale region-line primitive association features," *Journal of Photogrammetry and Remote Sensing*, vol. 123, pp. 104-113, 2017.

[2] Y.G. Liu, and S. H. Peng, "A new image compression algorithm base on rotating mapping and curve fitting," *Proc. 12th IEEE Int. Conf.* Signal Process., 2014.

[3] M. Pedro, H. Araujo, and N. Goncalves, "Pose estimation for general cameras using lines," *IEEE Trans. Cybern*., vol. 45, no. 10.pp. 2156-2164, 2015.

[4] V. C. P. Hough, "Method and means for recognizing complex patterns," U.S. Patent No. 3,069,654. 18 Dec. 1962.

[5] L. Xu, O. Erkki, and K. Pekka, "A new curve detection method: randomized Hough transform (RHT)," *Pattern Recognition Letters*, vol. 11, no. 5, pp. 331-338, 1990.

[6] D. P. Zhang, Y. S. Li, and Y. Liu, *et al*, "Randomized Hough Transform with Imitative Gradient Direction," *Computer Science*, vol. 33, no. 4, pp. 208-210, 2006.

[7] J. Matas, C. Galambos, and J. Kittler, "Robust detection of lines using the progressive probabilistic hough transform," *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 119-137, 2000.

[8] H. Freeman, "Boundary encoding and processing," *Picture Processing and Psychopictorics*, Orlando, USA: Academic Press, 1970.

[9] C. H. Teh, and R. T. Chin, "On the detection of dominant points on digital curves," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 8, pp. 859-872, 1989.

[10] S. Suzuki, and K. Abe, "Topological structural analysis of digitized binary images by border following," *Computer Vision, Graphics, and Image Process*, vol. 30, no. 1, pp. 32-46, 1985.

[11] M. Chen, Z. G. Cheng, and Y. C. Liu, "A robust algorithm of principal curve detection," *Proc. 17th IEEE Int. Conf. Pattern Recognition*, 2004.

[12] Y. Li, and R. L. Stevenson, "Multimodal image registration with line segments by selective search," *IEEE Trans. Cybern.*, pp. 1-14, 2016.

[13] J. B. Burns, A. R. Hanson, and M. R. Edward, "Extracting straight lines," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 4, pp. 425-455, 1986.

[14] K. Philip, L. Kitchen, and E. M. Riseman. "A fast line finder for vision-guided robot navigation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 11, pp. 1098-1102, 1990.
[15] A. Desolneux, M. Lionel, and J-M Morel, "Meaningful alignments," *International Journal of Computer Vision*, vol. 40, no. 1, pp. 7-23, 2000.
[16] A. Desolneux, M. Lionel, and J-M Morel, "Computational gestalts and perception thresholds," *Journal of Physiology-Paris*, vol. 97, no. 2, pp. 311-324, 2003.
[17] A. Desolneux, (2008), *From gestalt theory to image analysis: a probabilistic approach*, [Online], Available: http://www.springer.com.
[18] R. G. Von Gioi, J. Jérémie, and R. Gregory, "Multisegment detection," *Proc. 14th IEEE Int. Conf. Image Process.*, 2007.
[19] R. G. Von Gioi, J. Jérémie, and J-M Morel, *et al*, "LSD: a fast line segment detector with a false detection control," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 4, pp.722-732, 2010.
[20] V. Pătrăucean, G. Pierre, and R. G. Von Gioi. "A parameterless line segment and elliptical arc detector with enhanced ellipse fitting," *Proc. 12th European Conf. Computer Vision*, 2012.
[21] Liu, Xilong, *et al*. "Intelligent line segment perception with cortex-like mechanisms." *IEEE Trans. Systems, Man, and Cybernetics: Systems*, vol. 45, no. 12, pp. 1522-1534, 2015.
[22] J. Tang, L. Xiao, X. Li, *et al*, "A local structural descriptor for image matching via normalized graph Laplacian embedding," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 410-420, 2016.
[23] X. H. Han, Y. W. Chen, and G. Xu, "High-order statistics of weber local descriptors for image representation," *IEEE Trans. Cybern*., vol. 45, no. 6, pp. 1180-1193, 2015.
[24] T. Serre, L. Wolf, and S. Bileschi, *et al*, "Robust object recognition with cortex-like mechanisms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 3, pp. 411-426, 2007.
[25] L. F. Bo, X. F. Ren, and D. Fox, "Hierarchical matching pursuit for image classification: Architecture and fast algorithms," *Proc. 25th Annual Conf, Neural Information Processing Systems*, 2011.
[26] A. Coates, and Y. Ng. Andrew, "Selecting receptive fields in deep networks," *Proc. 25th Annual Conf. Neural Information Processing Systems*, 2011.
[27] D. R. Martin, C. F. Charless, and M. Jitendra, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 5, pp. 530-549, 2004.
[28] D. Martin, C. Fowlkes, D. Tal, *et al*. "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," *Proc. 8th IEEE Int. Conf. Computer Vision*, 2001.
[29] P. Arbelaez, M. Maire, C. Fowlkes, *et al*. "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898-916, 2011.
[30] C. Van Rijsbergen, *Information Retrieval*, 2nd ed. Dept. of Computer Science, Univ. of Glasgow, 1979.
[31] B. Xiao, H. G. Zhang, and J. Zhou, "VHR object detection based on structural feature extraction and query expansion." *IEEE Trans. Geosci. Remo. Sens.*, vol. 52, no.10, pp. 6508-6520, 2014.
[32] H. G. Zhang, B. Xiao and J. Zhou, *et al*., "Object detection via structural feature selection and shape model." *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 4984-4995, 2013.
[33] S. Liu, and B. Xiao, "Discriminative features for image classification and retrieval." *Pattern Recognition Letters*, vol. 33, no. 6, pp. 744-751, 2012).
[34] Y. L. Guo, K. Naman, N. Maruthi, and K. Benjamin, "A multi-stage approach to curve extraction." *Proc. 13th IEEE Euro. Conf. Computer Vision*, 2014.
[35] M. Kolomenkin, I. Shimshoni, A. Tal, "Multi-Scale Curve Detection on Surfaces", *Proc. 26th IEEE Int. Conf. Computer Vision and Pattern Recognition*, 2013.

Mingyi Zhang received the B.Sc. degree from China University of Geosciences (Beijing), Beijing, China, in 2014, in electrical engineering and automation. She is currently working towards Ph.D. degree at the Institute of Automation, Chinese Academy of Sciences (IACAS), Beijing, China, in control science and engineering. Her current research interests include image processing and visual control.

Xilong Liu received the B.S. degree from Beijing Jiaotong University, Beijing, China, in 2009 and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, in 2014. He is currently an Associate Professor with the Re-search Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences (IACAS). His current research interests include image processing, pattern recognition, visual measurement and visual scene cognition.

De Xu (M'05–SM'09) received the B.Sc. and M.Sc. degrees from the Shandong University of Technology, Jinan, China, in 1985 and 1990, respectively, and the Ph.D. degree from Zhejiang University, Hangzhou, China, in 2001, all in control science and engineering. He has been with Institute of Automation, Chinese Academy of Sciences (IACAS) since 2001. He is currently a Professor with the Research Center of Precision Sensing and Control, IACAS. His current research interests include robotics and automation such as visual measurement, visual control, intelligent control, welding seam tracking, visual positioning, microscopic vision, and micro-assembly.

Zhiqiang Cao (SM'14) received the B.S. and M.S. degrees from the Shandong University of Technology, Jinan, China, in 1996 and 1999, respectively, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2002. He is currently a Professor with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences (IACAS). His current research interests include embedded vision and visual scene cognition, multi-robot systems, and net-worked robotic system.