Extracting Lineage Information from Hand-Drawn Ancient Maps

Ehab Essa[1], Xianghua Xie[2](✉), Richard Turner[3], Matthew Stevens[3], and Daniel Power[3]

[1] Department of Computer Science, Mansoura University, Mansoura, Egypt
[2] Department of Computer Science, Swansea University, Swansea, UK
x.xie@swansea.ac.uk
[3] Department of History and Classics, Swansea University, Swansea, UK
http://csvision.swan.ac.uk

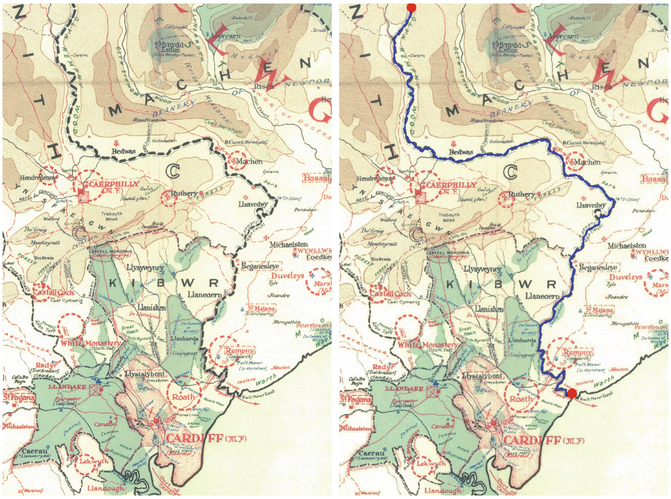
Abstract. In this paper, we present an efficient segmentation technique that extracts piecewise linear patterns from hand-drawn maps. The user is only required to place the starting and end points and the method is capable of extracting the route that connects the two, which closely colocates with the hand-drawn map. It provides an effective approach to interactively process and understand those historical maps. The proposed method employs supervised learning to evaluate at every pixel location the probability that such a lineage pattern exists, followed by shortest path segmentation to extract the border of interest.

1 Introduction

This work is concerned with processing ancient hand-drawn Welsh Marcher lordships maps to extract piecewise lineage patterns that corresponds to lordship boundaries. This system of landownership was established by the Norman kings after their conquest of England in 1066 to protect their borders with Wales. It reached its full extent after the English conquest of Wales in 1282. The basis for any historical or geographical study of the Welsh Marches relies on William Rees' four map sheets of South Wales and the Border in the Fourteenth Century (Ordnance Survey, 1933). The challenge is to use these maps as the basis for a geographical information systems, in which the extent and the ownership of each Marcher lordship can be depicted throughout their individual histories.

Manual segmentation of the historical hand-drawn maps is a time-consuming, tedious process, particularly when dealing with large maps. The maps are richly annotated and textured, e.g. various map keys, texts and colouring, see Fig. 1 for an example. The delineation and separation of individual elements play an important role in automated analysis and understanding of these maps. However, automated segmentation of these maps is a non-trivial task. Image segmentation can be largely classified into automatic and user-aided or semi-automatic approaches. In automatic segmentation methods, the segmented boundaries are found without any user interaction. These methods are typically working by

A. Campilho and F. Karray (Eds.): ICIAR 2016, LNCS 9730, pp. 268–275, 2016.
DOI: 10.1007/978-3-319-41501-7_30



(a) (b)

Fig. 1. An example of the South Wales lordship map. (a) original image. (b) final border of interest highlighted by blue. The two red dots are placed by the user and the path of interest is automated extracted. (Color figure online)

exploiting visual cues from the image, such as colour, texture, edge, and/or statistical distributions of those features as in [3,10,12–14,16]. However, in many cases they require high level knowledge to constrain the segmentation in order to cope with ambiguities in appearance, texture and shape. Semi-automatic segmentation methods introduce user interaction as an effective and efficient approach to transfer domain knowledge into the segmentation process. User can for example determine the region of interest, initialise the contour location [15,17], place some seed points on the object boundary [8,9] to guide the segmentation, or draw strokes inside and/or outside object of interest [1]. Generalising priors using learning techniques have been shown effective in many segmentation applications, e.g. [6].

In this paper, a semi-automatic segmentation method, with minimal user interaction, for extracting lineage patterns from ancient maps is proposed. The proposed method combines a machine learning method with a shortest path approach to delineate the boundaries of domains of marcher lords. Random forest classifier is used to generate a probability map of the potential location of the boundaries. A shortest path method based on the Dijkstra's algorithm is used to refine the segmentation between two seed points identified by the user. The proposed method is tested on the South Wales lordship maps in the fourteenth century as shown in Fig. 1(a) and achieves a promising results as in Fig. 1(b).

2 Proposed Method

Briefly, the proposed method first caries out pre-processing and feature extraction using histograms of orientated gradients (HoG) and histograms of intensities to highlight meaningful features for the boundary of interest. Next, random forest classifier is used to generate a probability map for the boundaries and suppress the background noise. The segmentation is then refined by finding the shortest path between user input seed points to obtain the final border.

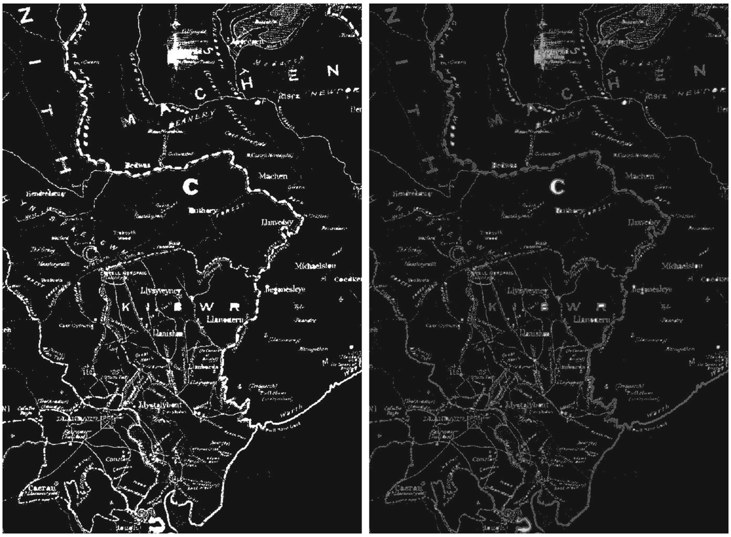
2.1 Pre-processing and Feature Extraction

The images contains various features that are of no interest to us for extracting those lordship boundaries. The borders of interest are piecewise linear and have reasonably good contrast. We thus devise a simple pre-processing step to reduce certain amount of background ambiguities, while maintaining as much as possible for the boundaries. We use a thresholded map to enhance the original image and to produce a greyscale image for further processing. The map images are first binarised in order to reduce the interference from other symbols and topography details with the boundaries. The binary image is obtained by applying a thresholding on the three colour channels of the image (red, green and blue). Since the value of each colour channel varies between 0 to 255, we retrieve the index of image pixels between 70 and 170 of each channel and combine the result to produce the binary map. Figure 2 shows an example of such binary image on the left and the equivalent greyscale image on the right. The greyscale image is obtained by multiplying the binary image with the original image and used to extract the histogram of image intensity features which is a set of pixel greyscale value quantised into a finite number of bins.

The boundaries of interest are made of short line segments, i.e. piecewise linear. We employ histogram of oriented gradients (HoG) as the second set of visual features to represent and capture those patterns. HoG is one of the most successful image features in object detection e.g. human detection [4]. It measures the distribution of edge directions in local segments of an image. The HoG descriptor is achieved by dividing the image into non-overlapping small regions known as cells, to compute the histogram of gradient orientations for each region. The concatenation of these histograms are then produced to represent the image features. Each local histogram is also normalised within a larger region (with overlapping) to enhance the invariance to contrast and illumination variations. Here, to compute the direction of edges, image gradient is computed by convolving the image with derivative filters. Each pixel within the cell votes for one of the 9 histogram bins. The contrast normalisation is carried out based on L2-norm.

2.2 Boundary Evaluation

Next, we train a supervised classifier to evaluate every each pixel location for the possibility of containing one of those lineage patterns. Various machine learning



(a) (b)

Fig. 2. Image pre-processing. (a) binary image. (b) greyscale image generated from the binary image and the original image. Feature extraction is carried out on the new greyscale image.

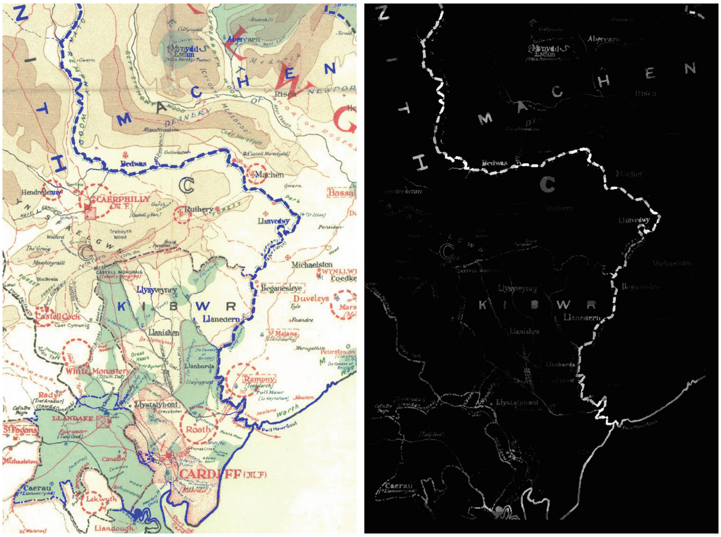
methods may be considered. Random Forest (RF) [2] is an ensemble-based classifier that combines a collection of decision trees to reduce the effect of overfitting and alleviate outlier problems to improve accuracy. RF is similar to bagging method in that it randomly samples training examples to train each decision tree separately. In addition, RF chooses a random subset of features attributes at each node to find the optimal tree splitting. In this work, we classify image pixels into two classes: boundary of the marcher lordship domains, and non-boundary.

In RF classification, a set of tress $r \in \{1, \cdots, R\}$ is trained separately on sub-sampling of the training examples. The testing of unseen parts of the map data is accomplished by introducing the unseen sample t to each decision tree starting from its root until it reaches the corresponding leaves and finally combines the predication result of each tree by computing the average predication of each class c:

$$P(c|v) = \frac{1}{R}\sum_{r=1}^{R} p_r(c|t). \tag{1}$$

Here, the number of trees R of the RF classifier sets to $R = 500$ and the number of attributes f randomly selected at each split is defined as $f = \sqrt{l}$, where l is the feature vector length.

The RF is trained on two features as aforementioned in Sect. 2.1, the histogram of the image intensity and HoG. These features extracted from a



(a) (b)

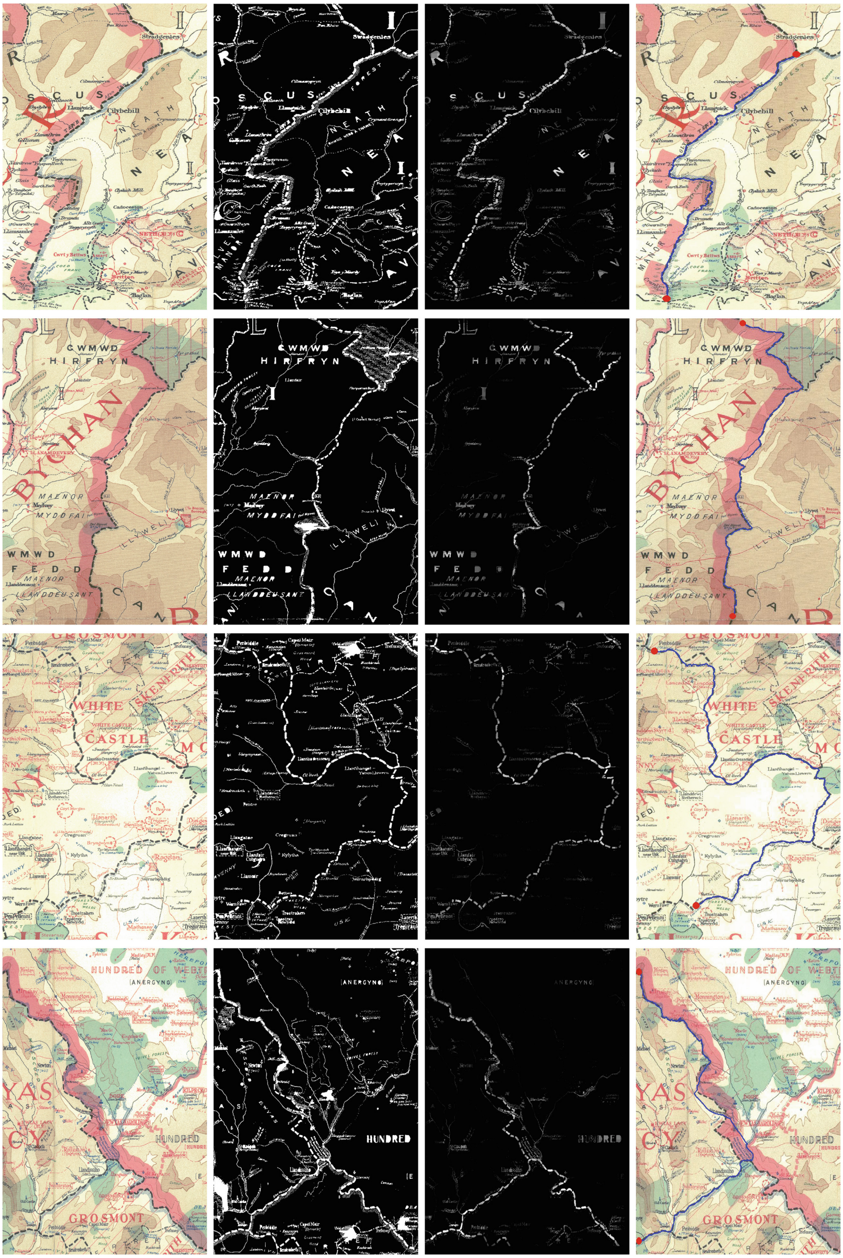
Fig. 3. The output of the RF classifier. (a) a binary classification result (highlighted by blue) superimposed on the original map. (b) the corresponding probabilistic map showing the possibility of each pixel in the map being considered as a pixel on the boundary of interest. The brighter the pixel the higher its probability. (Color figure online)

scanning window of size 128 × 128. An example binary classification result of the RF is shown in Fig. 3(a). The classifier can effectively highlight the border of interest. However, there are still some symbols that are misclassified, e.g. handwritten characters, which is inevitable.

2.3 Refinement Process Based on Shortest Path

Given that the boundaries of interest only contains a very small percentage of each entire map, eliminating the false positives while retaining high true positive rate is non-trivial. and may not be entirely plausible. In order to solve this problem, we proposed to use a shortest path algorithm based on Dijkstra's algorithm [5]. The proposed method finds the optimal path with minimum cost between two user points to refine the segmentation result. The cost function is defined using the probability values produced by the RF classifier. An example of the RF probability map is displayed in Fig. 3(b), where the brightness corresponds to the probability of the local region to be on the boundaries (i.e. bright white refers to high probability).

Shortest path is a combinatorial optimisation problem that working by finding the less costly solution of the problem from a finite set of all feasible solutions. The image is discretised to a graph $G(V, E)$, where V is a set of vertices represent



(a) (b) (c) (d)

Fig. 4. Segmentation of the South Wales marcher lordship map. (a) original image. (b) binary image. (c) RF probability map. (d) the final border (blue) between two user points (red) superimposed on the original map. (Color figure online)

image pixels, and E is the set of edges. These edges are connecting vertices and also, carrying a weight value that reflects the cost of using this path.

There are several methods available using the shortest path algorithm, e.g. Intelligent Scissors [11] and Live Wire [7] which segment the image on the fly by following the user clicks. The costs of edges are usually defined based on the gradient magnitudes and orientation. However these methods are sensitive to background noise.

In this work, the Dijkstra's algorithm is used to find the path between two vertices in a graph where the total associated edge costs are minimised. Dijkstra's algorithm is a single-source algorithm that locates the shortest path from the source vertex to all other vertices in the graph with non-negative edge weights. The algorithm creates a shortest path tree starting from the source vertex and iteratively adding the vertex that has the lowest distance to the source. Each vertex is exploring its unvisited vertex neighbours and updating its assign distance value. As long as the algorithm does not reach the destination vertex, it keeps picking up unvisited vertex with the lowest distance value to be the new current vertex and it repeat these steps. The cost of each edge is defined as the negative log of the RF's probability value. Figure 1(b) shows the result of using the shortest path algorithm, cf. Fig. 3. This approach allows user to efficiently extract boundary of interest with just placing two points at a time.

3 Results

Here, we briefly summarise the evaluation that took place and the results we have achieved. The proposed method was tested on the South Wales marcher lordship map in the fourteenth century. The dimension of the image is 24401 × 16610 with resolution 600 dpi and bit depth 24. The map was recreated by William Rees and published by the Ordnance Survey in 1932. Due to the large size of the map, it is segmented to relatively small blocks for testing. The training of the RF classifier carries out on two blocks of the map of size 2000 × 2000.

Figure 4 shows some examples of the segmentation result. The boundaries of the domain marcher lord are drawn as a dash line. As illustrated in Fig. 4(b), the binary image removes some of the unrelated details that have the colour distribution differ from that of those boundaries. RF probability maps and the final border segmentation superimposed on the original maps are shown in columns (c) and (d). The proposed method successfully highlights the boundaries of interest. A full quantitative evaluation requires manual annotation of all the borders which is very time consuming. However, the qualitative results demonstrate the potential of the method to be applied to effectively extract lineages from those historic maps. This method serves as an important step in fully dissecting and tokenising those maps.

4 Conclusions and Future Work

We presented a semi-automatic segmentation method for Welsh marcher lordships map in order to delineate the boundaries of the marcher lord domains.

The proposed method extracts local features based HoG to train an RF classifier in order to produce a probability map of the boundary location. The segmentation is refined by using a shortest path algorithm based on Dijkstra's method to highlight the final border. The experimental results show promising performance of the proposed method. In the future work, we will extend the method to segment the map keys such as chief and lesser castles, and the other types of features.

References

1. Boykov, Y., Funka-Lea, G.: Graph cuts and efficient N-D image segmentation. IJCV **70**(1), 109–131 (2006)
2. Breiman, L.: Random forests. Mach. Learn. **5**(1), 5–32 (2001)
3. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. IEEE T-PAMI **24**(5), 603–619 (2002)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. CVPR. **1**, 886–893 (2005)
5. Dijkstra, E.: A note on two problems in connexion with graphs. Numer. Math. **1**(1), 269–271 (1959)
6. Essa, E., Xie, X., Sazonov, I., Nithiarasu, P., Smith, D.: Shape prior model for media-adventitia border segmentation in IVUS using graph cut. In: Menze, B.H., Langs, G., Lu, L., Montillo, A., Tu, Z., Criminisi, A. (eds.) MCV 2012. LNCS, vol. 7766, pp. 114–123. Springer, Heidelberg (2013)
7. Falcão, A.X., et al.: User-steered image segmentation paradigms: live wire and live lane. Graph. Models Image Process. **60**(4), 233–260 (1998)
8. Jones, J.-L., Essa, E., Xie, X., Smith, D.: Interactive segmentation of media-adventitia border in IVUS. In: Wilson, R., Hancock, E., Bors, A., Smith, W. (eds.) CAIP 2013, Part II. LNCS, vol. 8048, pp. 466–474. Springer, Heidelberg (2013)
9. Jones, J.L., Xie, X., Essa, E.: Combining region-based and imprecise boundary-based cues for interactive medical image segmentation. IJNMBE **30**(12), 1649–1666 (2014)
10. Malik, J., Belongie, S., Leung, T., Shi, J.: Contour and texture analysis for image segmentation. IJCV **43**(1), 7–27 (2001)
11. Mortensen, E.N., Barrett, W.A.: Interactive segmentation with intelligent scissors. Graph. Models Image Process. **60**(5), 349–384 (1998)
12. Rotem, O., Greenspan, H., Goldberger, J.: Combining region and edge cues for image segmentation in a probabilistic gaussian mixture framework. In: CVPR, pp. 1–8 (2007)
13. Shen, W., Wang, X., Wang, Y., Bai, X., Zhang, Z.: Deepcontour: a deep convolutional feature learned by positive-sharing loss for contour detection. In: CVPR, pp. 3982–3991 (2015)
14. Xie, X., Mirmehdi, M.: Texture exemplars for defect detection on random textures. In: Singh, S., Singh, M., Apte, C., Perner, P. (eds.) ICAPR 2005. LNCS, vol. 3687, pp. 404–413. Springer, Heidelberg (2005)
15. Xie, X., Mirmehdi, M.: Magnetostatic field for the active contour model: a study in convergence. In: BMVC, pp. 127–136 (2006)
16. Xie, X., Mirmehdi, M.: TEXEMS: random texture representation and analysis. In: Handbook of Texture Analysis, Chapter 4 (2008)
17. Yeo, S., Xie, X., Sazonov, I., Mirmehdi, M.: Geometric potential force for the deformable model. In: BMVC (2009)