A Self-supervised Road Detection Method Based on Gabor Filter

Huayao Chang, Junzheng Wang, Jing Li

School of Automation, Beijing Institute of Technology, Beijing, China
changhy@bit.edu.cn

Abstract: Visual navigation is an important technology for Autonomous Guided Vehicles and road detection is the essential prerequisite. It is a challenging problem to distinguish the road or its boundaries in unstructured environments for the lack of discernible features. A self-supervised road detection method is proposed in this paper. Vanishing point is first detected using only four Gabor filters to precisely estimate the local dominant orientation at each pixel location, and an adaptive soft voting scheme is used to prevent tending to favor points that are high in the image. Training area is defined with vanishing point, which color feature is used for building self-supervised learning models to describe the road segment. Road patches are selected by measuring the roadness score. Experimental results have shown the efficiency of the method in terms of detection result and time saving.

Introduction

Visual navigation is an important technology for Autonomous Guided Vehicles and the road detection is the essential prerequisite. In general, roads in the real world can be divided into structured roads and unstructured roads. Up to until the present, various vision-based road-detection algorithms have been developed. Most of the early systems have focused on structured roads. Edges, intensities or other lane-marking features are commonly used for lane detection of structured roads, which have obvious lane markings, clear edges, relatively high intensities and specific colors and features. McCall et al. [1] proposed the VioLET system utilized steerable filters and performed an excellent survey of lane-detection researches.

For unstructured roads that have no obvious lane markings or lane boundaries, color and texture information combined with edges are often employed to distinguish the road surface from the surroundings under the assumption that the color or texture of the road surface is very different from the surroundings beside the road. Navlab [2] developed in CMU uses a number of Gaussian color models to represent the road and non-road colors (UNSCARF, SCARF). However, it requires some overlapping between the frames. Thus, this system is not convenient for suddenly changing road surfaces. Estimation of an optical flow [3] is often used but will fail on chaotic roads when the camera is unstable. Other methods attempt to use Hough transform [4], the main drawback of these methods are that they provide good performance only for roads with noticeable marking or borders.

Rasmussen [5] investigates the grouping of dominant orientations of a texture flow that is suitable for unstructured or ill-structured roads with no significant borders. Kong et al. [6] proposed the idea of a locally adaptive soft voting scheme to prevent tending to favor points that are high in the image. Ondrej et al. [7] proposed a method to detect the vanishing point based on the estimation of a texture flow and use color models for road segmentation. A bank of 2D Gabor filters with a large number of orientations is used to achieve a precise angular resolution for the local dominant orientation and it is inefficient. Peyman [8] proposed a fast vanishing point detection method to overcome this drawback.

A self-supervised road detection method is proposed in this paper. First, the vanishing point estimation is performed, which is used to detect the training area for learning of road color models. Then, self-supervised learning continues. Finally, road patches are selected by measuring of the roadness score.

Vanishing point estimation

In general, the vanishing point is defined by the intersection of the perspective projection of a set of lines that are parallel in world, but not parallel to the image plane, and is specified in image coordinates. The vanishing point plays an important role as a global constraint for detecting road direction, since all parallel border road lines, road edges, and even ruts and tire tracks left by previous vehicles on the road appear to converge into a single vanishing point. It is usually possible to detect this point by Hough transform, but such approaches usually completely fail in the case of unstructured roads.

Parameter selection of Gabor filter. Texture-based vanishing point detection approaches often apply a bank of oriented filters such as Gabor filter banks or steerable filter banks and choose the orientation corresponding to the maximum filter response as the dominant texture orientation at each pixel location. Later, each local dominant orientation votes for the location of road's vanishing points. A location with maximum votes is considered as the vanishing point of the road.

The dominant orientation θ at pixel $p = (x, y)$ of an image is the direction that describes the strongest local parallel structure or texture flow. 2D Gabor filter is used to get θ. If the set of $k \times k$ Gabor kernel with a known orientation φ_n and radial frequency $\omega_0 = 2\pi / \lambda$, the Gabor filters can be written as

$$\widehat{g}_{odd}(x, y, \varphi_n, \lambda) = \exp(-\frac{1}{8\sigma^2}(4a^2 + b^2))\sin(\omega_0 a)$$

$$\widehat{g}_{even}(x, y, \varphi_n, \lambda) = \exp(-\frac{1}{8\sigma^2}(4a^2 + b^2))\cos(\omega_0 a)$$

where $a = x\cos\varphi_n + y\sin\varphi_n, b = -x\sin\varphi_n + y\cos\varphi_n$,

$$\varphi_n = \frac{(n-1)\pi}{N_\varphi}, \quad n = 1, 2, ..., N_\varphi.$$

In order to estimate the dominant orientation $\theta(p)$ at each pixel location $p(x, y)$ in the image, the grayscale input image $I(x, y)$ is convolved with the bank of Gabor filters with predefined orientations. Finally, the Gabor energy $E(\varphi_n, \lambda)$ is calculated as the magnitude of the complex filter response as follows

$$E(\varphi_n, \lambda) = \sqrt{\begin{array}{l}[\widehat{g}_{odd}(x, y, \varphi_n, \lambda) * I(x, y)]^2 + \\ [\widehat{g}_{even}(x, y, \varphi_n, \lambda) * I(x, y)]^2\end{array}}$$

The orientation corresponding to the strongest Gabor energy response of all orientations at each pixel location is chosen as the local dominant texture orientation. Four parameters should be determined to calculate $E(\varphi_n, \lambda)$ and they are φ_n, ω_0, k and σ. Kong [6] considered 5 scales on a geometric grid and 36 orientations. Rasmussen [5] used a Gabor filter wavelength of

$$\lambda = 2^{\log_2(I_w)-5}$$

where I_w is the width of the road image. This provides good trade-off between computational complexity of multiscale schemes and a precision of a single scale bank of filters for most cases.

Accounting to the relationship of $\sigma = \pi / \omega_0$, σ can be determined with λ. Since most information of a Gabor wavelet is contained in the range of $[-3\sigma, 3\sigma]$, parameter k is taken as $k = 6\sigma + 1$. For 128×128 images, the parameter of Gabor kernel can be selected as $\lambda = 4, \sigma = 2$ and the size of filter template is 13×13.

Dominant orientation estimation. Different from the normally used number of set of orientations such as 72 or 36, four oriented Gabor filters $E_{\varphi_n}(p)$ is calculated the same as Peyman [8] with $\varphi_n \in \{0°, 45°, 90°, 135°\}$.

Gabor energy responses $E_{\varphi_n}(p)$ are sorted based on their magnitudes in descending order $E_{\varphi_1}(p) > E_{\varphi_2}(p) > E_{\varphi_3}(p) > E_{\varphi_4}(p)$. Then, the resulting vector of the two most dominant filter activation strengths will be used to represent the local texture orientation as follows:

$$V(p) = V_x(p) + jV_y(p) = \sum_{i=1}^{2} E_\varphi^i(p)e^{j\varphi_i} \tag{1}$$

where φ_1 and φ_2 represent the angle of the two dominant Gabor energy responses. The estimated dominant orientation is determined by

$$\hat{\theta}(p) = \tan^{-1}\frac{V_y(p)}{V_x(p)} = \frac{E_\varphi^1 \sin\varphi_1 + E_\varphi^2 \sin\varphi_2}{E_\varphi^1 \cos\varphi_1 + E_\varphi^2 \cos\varphi_2} \tag{2}$$

However, if the pixel is related to a feature with no apparent dominant orientation, the Gabor energy response values may be very similar for all four orientations. In such a case, relying only on the two strongest filter responses may result in a large estimation error of the dominant orientation. To solve this problem, two new vectors are introduced as follows:

$$\begin{cases} \|S_1(p)\| = E_\varphi^1(p) - E_\varphi^4(p) , & \varphi_{S_1}(p) = \varphi_1(p) \\ \|S_2(p)\| = E_\varphi^2(p) - E_\varphi^3(p) , & \varphi_{S_2}(p) = \varphi_2(p) \end{cases}$$

Where φ_{S_1} and φ_{S_2} is the angle of $S_1(p)$ and $S_2(p)$ respectively. Then, (1) can be replaced by these two new vectors as

$$V(p) = \sum_{i=1}^{2} S_i(p)e^{j\varphi S_i} \tag{3}$$

Actually, when a pixel has no apparent dominant orientation, we calculate the dominant orientation as follows:

$$\begin{cases} if\ E_{\varphi 1}(p) < T_{e1}\ and\ E_{\varphi 1}(p)/E_{\varphi 4}(p) < T_{e2},\ case\,1 \\ if\ E_{\varphi 1}(p) > T_{e1}\ and\ T_{e2} < E_{\varphi 1}(p)/E_{\varphi 4}(p) < T_{e3},\ case\,2 \\ else,\ case\,3 \end{cases}$$

In case 1, the pixel is excluded from voting scheme. In case 2, the dominant orientation is calculated with formula (3) and in case 3, the dominant orientation is calculated with formula (2).

Vanishing point voting. Once the dominant orientation is estimated at each pixel, a ray defined by $r_p = (p, \hat{\theta}_p)$ is drawn upward in an accumulator space, which has the same size as the input image. Every pixel lies on the ray r_p will get incremented by 1 provided that it is located above point $p(x, y)$. This voting strategy tends to favor the pixels higher in the image. So we just voting every ray with a radius set to be $r_1 = 0.35\sqrt{I_w^2 + I_h^2}$, where I_w and I_h are the width and height of image respectively.

Road detection

Vanishing points provides information about road direction and we can use it to build the training area. Color feature is then used for road segmentation and we chose algorithm based on Gaussian Mixture Models (GMM) and self-supervised learning.

Road color models. The training area is firstly built by the estimated vanishing point. It is initialized in its default position which is centered trapezoid at the bottom of image and then is shifted with the position of vanishing point. The offset can be calculated by

$$x_{offset} = \frac{(I_h - h)(v_x - I_{\frac{w}{2}})}{(I_h - v_y)} \tag{4}$$

where (v_x, v_y) is the coordinate of vanishing point and h is a constant, which can be set to half of the height of training area.

Once the training area is defined, the next step is the building of the Gaussian mixture models (GMM) in normalized RGB space, with k Gaussians to be found. Our method is to classify all pixels in the training area using k-means++ clustering, and then to model each cluster by its average value, its covariance matrix, and its mass which is defined as the total number of pixels in the cluster. In addition to the k training models, n_l learned models exist to represent the history road color features. At the beginning, all color models are null. Each training model is compared with learned models at any given frame of the incoming video stream. If the training models overlap with a learned model according to the following relation:

$$(\mu_L - \mu_T)^{\mathbf{T}}(\Sigma_L + \Sigma_T)^{-1}(\mu_L - \mu_T) \le d_1 \tag{5}$$

where μ is a mean vector, Σ is a covariance matrix and d_1 is a constant. The training model is regarded as new data for that learned model, and it is used to update the learned model. The updating formula is:

$$\mu_{updated} = \frac{m_L \mu_L + m_T \mu_T}{m_L + m_T} \ , \Sigma_{updated} = \frac{m_L \Sigma_L + m_T \Sigma_T}{m_L + m_T}$$

$$m_{updated} = m_L + m_T \tag{6}$$

where m is associated mass to the model. Otherwise, if all models are not full, the new model is created. If all models are full, then the model with the lowest mass is discarded and a new one is created in its place.

Road segmentation. Once all the models are updated with color models of training area, we are able to measure a degree of belonging to the road/non-road region of pixels outside the training area. All pixels of the image are assigned a "roadness" score, which is measured as a minimum of the Mahalanobis distance between each pixel and the n_b learned models with biggest mass.

$$D(\mathrm{p}, \mu_i) = \min_i((\mathrm{p} - \mu_i)^T)\Sigma_i^{-1}(\mathrm{p} - \mu_i)) \tag{7}$$

To extract only road segments, a threshold is set and its default value is determined by pixels belonging to the training area. If the score of a pixel is less than $H_d = \mu + 3\sigma$, which is the mean value and variance of score about pixels in training area, the pixel is labeled as road pixel.

Experimental Results

In this section, the proposed method was tested on the images provided by CMU image library (http://vasc.ri.cmu.edu/idb/html/road/index.html) and DARPA. The experiments are performed on the computer with Pentium CPU 2.8GHz and 1GB RAM. The program developing environment is Visual C++ 6.0 and Opencv.

Figure 1 is the vanishing point detection result. Red point is the vanishing point and green region is the approximate road area. The second row is the voting space.
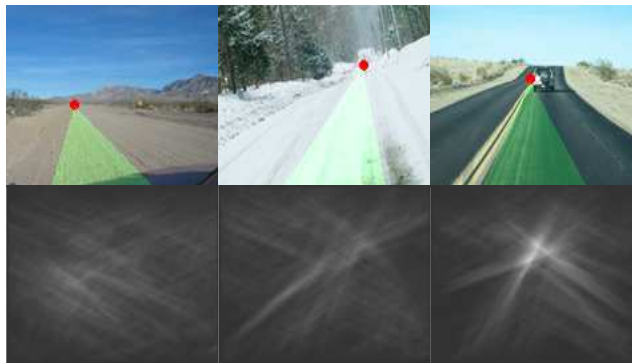

Fig. 1　Vanishing point detection

Figure 2 is the road detection result. Yellow region is the training area and the red region is the detected road area. The image resolution is 256×240 and the processing time of proposed algorithm is around 500ms. Executing Gabor filter takes nearly 420ms and it can be reduced by subsampling to speed up the whole algorithm.

Fig. 2   Road detection result

Conclusion

A self-supervised road detection method is proposed in this paper. Vanishing point is first detected using only four Gabor filters to precisely estimate the local dominant orientation at each pixel location. Training area is defined with vanishing point, which color feature is used for building self-supervised learning models to describe the road segment. Road patches are selected by measuring the roadness score. Experimental results have shown the efficiency of the method.

References

[1] J. C. McCall, M. M. Trivedi. Video-Based Lane Estimation and Tracking for Driver Assistance: Survey, System, and Evaluation [J]. IEEE Trans. on Intelligent Transportation Systems, 2006, 7(1):20-37.

[2] J. Crisman, C. Thorpe. Unscarf, a color vision system for the detection of unstructured roads [C]. IEEE International Conference on Robotics and Automation, 1991, 2496– 2501.

[3] A. Wedel, T. Pock, C. Zach, etc. An improved algorithm for tv-l1 optical flow [C]. Statistical and Geometrical Approaches to Visual Motion Analysis: International Dagstuhl Seminar, 2009, 23-45.

[4] J. R. Claudio, K. R. Christian. Lane following and lane departure using a linear-parabolic model [J]. Image and vision computing, 2005, 23:1192-1202.

[5] C. Rasmussen. Grouping dominant orientations for illstructured road following [C]. Proc. IEEE Comp. Soc. Conf. on Computer Vision and Pattern Recognition. 2004, 470-477.

[6] H. Kong, J. Audibert, J. Ponce. General road detection from a single image [J]. IEEE Trans. Image Process, 2010, 19(8):2211-2220.

[7] M. Ondrej, P. Petr, Z. Ludek, etc. Robust detection of shady and highlighted roads for monocular camera based navigation of UGV [C]. IEEE International Conf. on Robotics and Automation, 2011, 64-71.

[8] M. Peyman, S. A. Janusz, W. S. Wijesoma. Fast vanishing-point detection in unstructured enviroments [J]. IEEE trans. on Image Processing, 2012, 21(1):425-430.