Robust Detection of Shady and Highlighted Roads for Monocular Camera Based Navigation of UGV

Ondrej Miksik, Petr Petyovsky, Ludek Zalud and Pavel Jura

***Abstract*— This paper addresses the problem of UGV navigation in various environments and lightning conditions. Previous approaches use a combination of different sensors, or work well, only in scenarios with noticeable road marking or borders. Our robot is used for chemical, nuclear and biological contamination measurement. Thus, to avoid complications with decontamination, only a monocular camera serves as a sensor since it is already equipped. In this paper, we propose a novel approach - a fusion of frequency based vanishing point estimation and probabilistically based color segmentation. Detection of a vanishing point, is based on the estimation of a texture flow, produced by a bank of Gabor wavelets and a voting function. Next, the vanishing point defines the training area, which is used for self-supervised learning of color models. Finally, road patches are selected by measuring of the roadness score. A few rules deal with dark cast shadows, overexposed highlights and adaptivity speed. In addition to the robustness of our system, it is easy-to-use since no calibration is needed.**

I. INTRODUCTION

Our robotic research group works on Orpheus-AC military reconnaissance mobile robot [1], [2]. The robot is a part of an armored vehicle for chemical, nuclear and biological contamination measurement (see Fig. 1). Its primary task, is to make the measurement and identification in areas with the highest risk of massive contamination. The robot is being developed for Czech Army. Although the robot is primarily teleoperated, we are working on autonomous functions, that will help the user to achieve higher universality and reliability in missions.

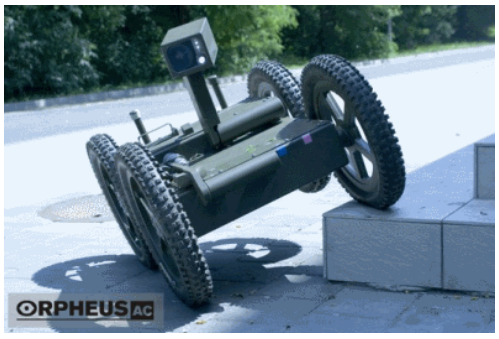


Fig. 1. Orpheus-AC robot

This project was supported by the Ministry of Education of the Czech Republic under Project 1M0567.

This research has been supported by the Czech Ministry of Education in the frame of MSM 0021630529 Research Intention Intelligent Systems in Automation.

Authors are with the Department of Control and Instrumentation, Faculty of Electrical Engineering and Communication, Brno University of Technology, Kolejni 4, Brno, Czech Republic. ondra.miksik@gmail.com, {petyovsky, zalud, jura}@feec.vutbr.cz

Fig. 2. Output of our system (best viewed in color)

One of the important missions planned for the new version of Orpheus-AC, can be described as follows: the robot moves about 100m ahead of the armored vehicle, while making the primary measurement of chemical contamination and radiation. Since the robot, together with accompanying vehicle, may move relatively rapidly (about 15 km/h) even in quite hard terrain, it may be difficult, or even impossible, for the operator to directly control the robot in the moving vehicle. For this reason, we plan to use a system that would be able to automatically control the robot's movement in order to follow the road. Several important features and demands of the system come from the description of the mission:

- It should be able to operate under a wide spectrum of operational conditions regarding climate, and surrounding environment - the system has to reliably find the way in diverse light conditions, like in direct sunlight, overcast, sunset, etc.
- It should work reliably on both high-quality roads as well as on roads barely visible even for humans including sand, concrete, tarmac, gravel, etc.
- It should use a minimum number of sensors - since the robot is intended to work in contaminated areas, it has to be extremely easy-to-decontaminate. Every irregularity on robot's surface means a serious problem. Since the robot is teleoperated, it is already equipped with high quality camera, so it appears as an obvious source of data.

Although the described scenario represents the primary practical problem, we solve the hereinafter by the described algorithms, we can foresee many other applications, both military and non-military, that would benefit from it. We

can name automatic road navigation for goods delivery (e.g. DARPA Grand Challenge), automatic return from teleoperated mission in case of signal loss, automatic mine/booby-trap detection, etc.

II. RELATED WORKS

A. Vision-based Road Segmentation

Many papers about vision-based road segmentation have been published during the last two decades. Most of the early systems have focused on structured roads. A well known project was developed in Carnegie Mellon University (CMU)'s Navlab [3], [4], that uses a number of Gaussian color models to represent the road and non-road colors (*UNSCARF*, *SCARF*). This navigation system is considered as a powerful, because it deals with both intersections and shadows, however, it requires some overlapping between the frames. Thus, this system is not convenient for suddenly changing road surfaces. A similar project based on stereo vision is named *ARGO* from Università di Parma [5], [6]. Another project from CMU Navlab called *ALVINN* deals with both, structured and unstructured roads, nonetheless an artificial neural network classifier is used, which means that it requires previously learned road models [7].

Other approaches focused on the optical flow estimation [8], [9] provide adaptive segmentation of the road area, but such methods do not work well on chaotic roads, when the camera is unstable and the optical flow estimation is not robust enough. Vehicle *Stanley* developed by Stanford AI Lab, successfully used the combination of laser range finders and camera [10]. It should be noted that commonly used outdoor lidars, are inappropriate for our mission because of a variety of reasons, like dimensions, weight, price, complications with decontamination, etc. Moreover, shadows are considered as non-road areas. Other methods attempt to use Hough transform [11], stereo vision [12], or radar [13]. The main drawback of these methods are, that they provide good performance only for roads with noticeable marking or borders.

For unstructured, or ill-structured roads with no significant borders, methods exist developed by the University of Delaware Dynamic Vision Lab, focused on the estimation of the vanishing point [14], [15]. They provide only information about course, but do not provide any information about free distance ahead of the robot.

B. Shady and Highlighted Roads

The essential problem of outdoor computer vision is color constancy. The importance of dealing with shadows and overexposed highlights, even rise up when the camera is moving. The elementary and widely used solution, is the processing of image in an alternative color system, like HSV/HLS or Lab/Luv instead of standard RGB. It is believed that segmentation in such color spaces is performed better, since brightness information is encoded in a different channel than chromatic information. However it was observed, that it performs well only in small variances, while performance against dark cast shadows and overexposed highlights is low. Moreover, a problem problem exists with measuring color distances, because Hue is represented as a color wheel (e.g. 2° and 358° are similar hues, but are numerically far away). Another ordinarily used system is an opponent color space, which exists in many modifications. It is widely used in a domain of local invariant feature detectors and descriptors [16], however it was observed, that its performance for road segmentation is poor.

Many algorithms are recently focused on *shadow removing* [17], [18]. The main drawbacks of these techniques are number of assumptions and constraints, because it relies on difficult modeling of physical properties of light and cameras. Hence, such techniques are not robust and work only in limited situations. On the other hand, there is significant progress in brightness and gamma invariant systems [19], [20].

III. VISION SYSTEM DESIGN

This paper, introduces a novel approach to robust detection of shady and highlighted roads by a monocular camera. By comparison with recently presented state-of-the-art methods, [10], [12], we neither use a laser range finder, nor stereo vision for extraction of the training area. Our system, is based on vanishing point estimation and does not need any time consuming calibration, or difficult classifier training or other sensors. Our approach is a fusion of the frequency based estimation of so called **vanishing point** and probabilistically based **texture segmentation**.

A combination of two different approaches, allows us to solve difficult situations without any a priori knowledge of robot's environment. The basic idea of our solution is estimation of the vanishing point, which determines the training area for texture segmentation. Next, road color models are constructed from sample pixels defined by the training area. These models are associated with previously learned models, which are stored in a memory. Further, learned models are adaptively updated. Therefore, the models include both the road colors' history and the current road appearance. A few simple rules define properties of the color segmentation system, like adaptivity speed, selectivity, robustness or behavior in shady and/or overexposed highlighted road segments.

The strategy of our vision system is the following: start with the vanishing point estimation, which is used to detect the training area for self-supervised learning of color models. Next, self-supervised learning continues, however, it is possible to perform road segmentation based on these models. Besides, a combination of two different approaches is advantageous, because in situations like sudden road texture or illumination change, we are still able to estimate the correct course, because if the color models are not consistent with current road surface, it is possible to use a vanishing point until new color models are learned.

IV. VANISHING POINT ESTIMATION

Parallel lines in the real world, do not look like parallel lines under the perspective projection. Therefore, borders of each straight road in an image plane, converge at some

point, the so called **vanishing point**. For well engineered structured roads, it is usually possible to detect this point by a "cascaded" Hough transform, however, such approaches usually completely fail in the case of unstructured roads.

A. Texture Flow Estimation

The first step of a vanishing point estimation algorithm is, the estimation of a *texture flow* (see Fig. 4). The dominant orientation $\theta(\mathbf{p})$ of an image at pixel $\mathbf{p}(x,y)$ describes strongest local parallel structure or texture flow. Various techniques exist, which can be used for estimation of dominant orientation, involving usage of Gaussian pyramids with principle component analysis, steerable filters, etc. Our approach is based on a bank of 2D Gabor wavelet filters since they are known to be accurate [14], [15].

Gabor wavelet filters are quite similar to the 2D receptive field profiles of the mammalian cortical simple cells, and shows suitable characteristics of spatial locality and orientation selectivity [21]. Gabor transformation is a special case of the Short Time Fourier Transform (STFT) which uses windows to determine the frequency and the phase content of the local parts of a signal as it changes over time. It was observed that Gaussian window provides the best trade-off between the product of a time period and bandwidth. Consequently, the 2D Gabor function is a product of an elliptical Gaussian and a complex plane wave. The Gabor wavelets are self-similar, which means that all kernels can be constructed from one mother wavelet by its dilation and/or rotation [22].

The set of $k \times k$ Gabor kernels for an orientation θ, wavelength λ and odd or even phase, the filters are defined by

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

where $x = y = 0$ is the kernel center. Next, a and b are defined as

$$\begin{aligned} a &= x\cos(\theta) + y\sin(\theta), \\ b &= -x\sin(\theta) + y\cos(\theta). \end{aligned} \quad (2)$$

Parameter σ is set as $\sigma = \frac{k}{9}$ and size of kernel k is determined by wavelength as $k = \frac{10\lambda}{\pi}$. To obtain even kernel "sin" is simply substituted by "cos"

$$\widehat{g}_{even}(x,y,\theta,\lambda) = \exp\left(-\frac{1}{8\sigma^2}(4a^2+b^2)\right)\cos\left(\frac{2\pi a}{\lambda}\right), \quad (3)$$

and other parameters are the same.

Then, $\widehat{g}$'s DC component is subtracted from Gabor kernel, to satisfy one of the design constraints for filters measuring phase disparities to ensure optimal phase behavior [23]

$$\widehat{g}_{DC}(x,y,\theta,\lambda) = \widehat{g}(x,y,\theta,\lambda) - \frac{1}{k^2}\sum_{x=-k/2}^{x=k/2}\sum_{y=-k/2}^{y=k/2}\widehat{g}(x,y,\theta,\lambda). \quad (4)$$

Finally, kernel's coefficients are normalized to make the filter more robust to spurious noise, so that L^2 norm is equal to one

$$\widehat{g}_{L_2}(x,y,\theta,\lambda) = \frac{\widehat{g}_{DC}(x,y,\theta,\lambda)}{\sqrt{\sum_{x=-k/2}^{x=k/2}\sum_{y=-k/2}^{y=k/2}\widehat{g}_{DC}(x,y,\theta,\lambda)^2}}. \quad (5)$$

Let $I(x,y)$ be the intensity value of a grayscale image at spatial coordinates (x,y). By convolution of an image I with each of n evenly spaced Gabor filter orientations, a square norm of the so-called *Gabor energy* (complex response) is computed to get the best characteristics of a local texture jet

$$\begin{aligned} \mathbf{E}(\theta,\lambda) =& [(\widehat{g}_{odd}(x,y,\theta,\lambda) * I(x,y)]^2 + \\ &+ [(\widehat{g}_{even}(x,y,\theta,\lambda) * I(x,y)]^2, \end{aligned} \quad (6)$$

where $*$ denotes convolution.

For n even and odd pairs of Gabor filters (e.g. $n = 36$ for the equidistantly spaced angle between 0° and 180°), the dominant orientation at pixel $\mathbf{p}(x,y)$ is chosen as the filter orientation which elicits the maximum complex response at that location

$$\theta_{max} = \arg\max_{\theta} \mathbf{E}(\theta,\lambda). \quad (7)$$

For an efficient computation, it is possible to apply convolution theorem and thus, filter's response can be computed as

$$\begin{aligned} \mathbf{E}(\theta,\lambda) =& [\mathscr{F}^{-1}\{\mathscr{F}\{\widehat{g}_{odd}(x,y,\theta,\lambda)\}\mathscr{F}\{I(x,y)\}\}]^2 + \\ &+ [\mathscr{F}^{-1}\{\mathscr{F}\{\widehat{g}_{even}(x,y,\theta,\lambda)\}\mathscr{F}\{I(x,y)\}\}]^2, \end{aligned} \quad (8)$$

where $\mathscr{F}$ denotes Fourier transform and $\mathscr{F}^{-1}$ inverse Fourier transform, respectively. Fourier transforms of Gabor filters can be precomputed by FFTW [24] and stored in a memory.

The last parameter which needs to be determined is a wavelength λ. With a priori knowledge of road texture wavelengths, camera extrinsic and intrinsic parameters, a bank of appropriate filters can be built and used for different parts of an image, however, it was reported that in general, a single wavelength, which can be computed according to a formula $\lambda = 2^{\log_2(I_w)-5}$ where I_w is the width of an image I, provides good trade-off between computational complexity of multiscale schemes and a precision of a single scale bank of filters for most cases (see Fig. 3).

Reliable estimation of a dominant orientation, is important to ensure a valid sharp peak for voting function. Usually, (sub)urban environments contain many artifacts, which negatively influences vanishing point estimation. Hence, the input image is firstly smoothed by Gaussian filter. Moreover, to deal with difficult illumination conditions, experiments

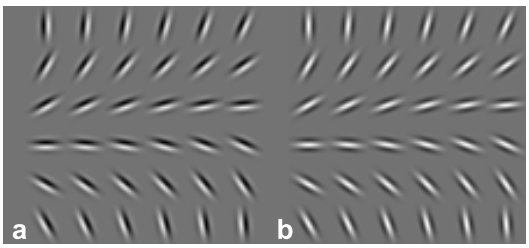


Fig. 3. Odd (a) and even (b) Gabor filters for n=36, $\lambda = 5$, $k = 16$, $\sigma = \frac{k}{9}$.

showed, that it is better to estimate dominant orientation in the B channel of RGB color system instead of pure grayscale.

B. Vanishing Point Voting

The second stage of the vanishing point estimation is voting [14]. For sake of simplicity, we assume, that in an ideal case of a pinhole camera, each straight road which is distinguished by parallel lines, has only one unique vanishing point in the image plane, due to the perspective projection. In the real world, this assumption is not a constraint, since most types of (curved) roads (excluding crossroads) are projected from the base plane in a similar way to the image plane, or we simply estimate the strongest vanishing point.

All vanishing point candidates are defined by a region C (discussed below). The set of possible vanishing points for each pixel $\theta_{max}(\mathbf{p})$ with dominant orientation θ_{max} are all pixels along the line defined by $(\mathbf{p},\ \mathbf{p}(\theta_{max}))$. In fact, angular resolution of dominant orientation estimation in the previous step has a finite value of $\frac{\pi}{n}$. Let the angle of the line joining an image pixel $\mathbf{p}$ and a vanishing point candidate $\mathbf{v}$ is $\alpha(\mathbf{p},\mathbf{v})$, then $\mathbf{p}$ votes for $\mathbf{v}$ if the difference between $\alpha(\mathbf{p},\mathbf{v})$ and $\theta_{max}(\mathbf{p})$ is within the dominant orientation estimator's angular resolution (coefficient $\gamma = 2$ sets selectivity).

$$vote(\mathbf{p},\mathbf{v}) = \begin{cases} 1 & \text{if} |\alpha(\mathbf{p},\mathbf{v}) - \theta_{max}(\mathbf{p})| \leq \frac{\gamma\pi}{n}, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

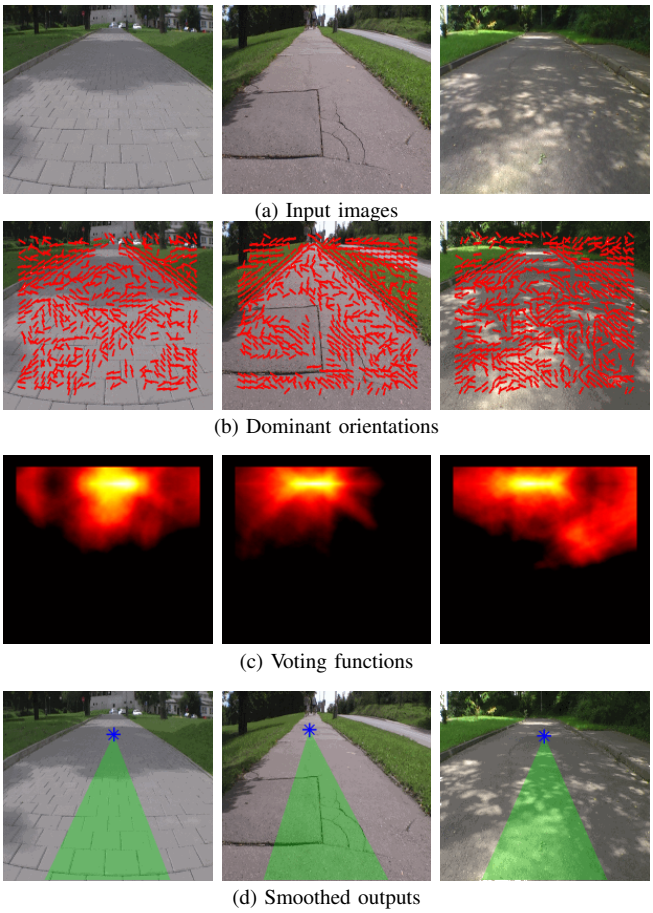
Next, the definition of an objective function for each vanishing point candidate $\mathbf{v}$ is straightforward

$$votes(\mathbf{v}) = \sum_{\mathbf{p}\in R(\mathbf{v})} vote(\mathbf{p},\mathbf{v}), \tag{10}$$

where $R(\mathbf{v})$ is a voting region, which includes all image pixels below the horizontal line $\mathbf{l}$ determined by the current vanishing point candidate $\mathbf{v}$, minus edge pixels excluded from convolution by the kernel size.

Finally, we discuss region C, which defines the vanishing point candidates. For rural roads without any high obstacles, it is defined as a top $3/4$ of the image. In the case of a rural road, it is usually possible to detect the horizon [25] and limit the top boundary by this line. Restriction of region C saves computational complexity and leads to better accuracy.

In the case of (sub)urban roads, there arises problems with many obstacles, which produce many false dominant orientations, because we are not able to decide, whether dominant orientation at pixel $\mathbf{p}$ is produced by road texture, or by any obstacle. Usually, this leads to misidentification of the true vanishing point. To overcome troubles in a (sub)urban environment, we set the camera tilt so that approximately 75% of the image is created by the road. Horizon detection is usually imposible or useless in an urban environment, because sky creates only a few top rows of the image. Thus, the top boundary of the region C is manually fixed to some assumed value C_{top} (identical with the camera position). Besides, all pixels with almost horizontal or vertical dominant



(a) Input images

(b) Dominant orientations

(c) Voting functions

(d) Smoothed outputs

Fig. 4. Vanishing point estimation

orientations θ_{max} are rejected before we start the voting

$$\theta_{max}(\mathbf{p}) = \begin{cases} \theta_{max}(\mathbf{p}) & \text{if } \frac{(90l+5)\pi}{180} < \theta_{max}(\mathbf{p}) < \frac{(90l+85)\pi}{180}, \\ \text{rejected} & \text{otherwise,} \end{cases} \tag{11}$$

where $l = \{0,1\}$.

C. Smoothing

One can see, that extraction of a vanishing point from objective function is straightforward - it is pixel, where the number of votes elicits its maximum. Instead of usage of output independently per each frame, we rather run a smoothing filter throughout the whole sequence to reduce influence of noise and to avoid the jumpy characteristic of output. Particle filters (sequential Monte Carlo) are often used in computer vision since they overcome many limiting assumptions of Kalman Filters.

Particle filters are successful in the tracking of multimodal distributions. Unfortunately, objective function in an urban environment usually do not have sharp a maximum which is necessary for correct prediction. Thus, a DC component is subtracted from objective function

$$V_{DC}(x,y) = V(x,y) - \frac{\sum\sum_{a,b\in V} V(x,y)}{(I_w - k)(I_h - k)}, \tag{12}$$

where V(x,y) denotes voting function. Negative values which are introduced by this subtraction are removed

$$V_{DCcorr}(x,y) = \begin{cases} V_{DC}(x,y) & \text{if} \quad V_{DC}(x,y) > 0, \\ 0 & \text{otherwise}, \end{cases} \tag{13}$$

After this preprocessing, a standard CONDENSATION algorithm proposed by Isard and Blake [26] is performed and the best estimate of the state of the system $\mathbf{x}_t$ in time t is given by

$$\mathbf{x}_t = \sum_{i=1}^{i=N} \pi_{t,i} \mathbf{s}_{t,i}, \tag{14}$$

where $\mathbf{s}_{t,i}$ represents samples (particles) in time t, N is a number of particles and $\pi_{t,i}$ are their associated weights.

V. ROAD EXTRACTION

In fact, a vanishing point does not tell us anything about a road surface. Vanishing points provides information about direction, however we do not have any information about free space ahead of the robot. Thus, another algorithm based on adaptive color segmentation is needed. We chose algorithm based on Gaussian Mixture Models (GMM) and self-supervised learning (see Fig. 6).

A. Training Area

By comparison with previously published algorithms [10], [12], the training area is determined by the estimated vanishing point. The training area is initialized in its default position - centered trapezoid at the bottom of the image. Next, to remove non-road regions, the training area is shifted

$$x_{offset} = \frac{(I_h - h)(v_x - I_{\frac{w}{2}})}{I_h - v_y}, \tag{15}$$

where I_h is image height, $I_{\frac{w}{2}}$ is half of image width, v_x and v_y are spatial coordinates of vanishing point and h denotes projection constant, which is set to the half of height of a training area.

After transition of the default training area to the new position, two regions are settled. The first one ($area_1$) is delimited by lines joining the vanishing point $\mathbf{v}_p$ and ending points of the polygon's base, the second one ($area_2$) is created by lines joining the vanishing point with bumpers (approx. 10 pixels from image boundaries). The final shape of the training area is computed as an intersection of $area_1$ and $area_2$ (see Fig. 5)

$$area = area_1 \cap area_2. \tag{16}$$

B. Color Models Management

In this section, we describe handling with color models, which are learned from samples defined by the training area. GMM based segmentation can be performed in an arbitrary color space. The experiments showed, that segmentation based on RGB color space works well, however, if environment allows us to use less selective color space, we strongly recommend brightness-invariant $c_1c_2c_3$ color space [20] which successfully deals with uncertain illumination.

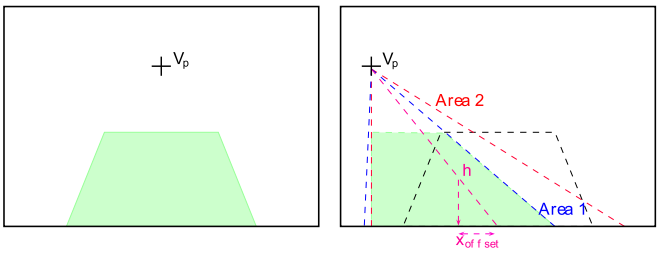


Fig. 5. Training area in its default position (left) and shifted training area (right).

$$c_1c_2c_3 = \begin{cases} c_1 & = \arctan \frac{r}{\max(g,b)}, \\ c_2 & = \arctan \frac{g}{\max(r,b)}, \\ c_3 & = \arctan \frac{b}{\max(r,g)}. \end{cases} \tag{17}$$

1) Construction: Once the training area is defined, the next step is the building of the Gaussian mixture models (GMM), which are used to detect the road outside of the training area. Instead of a commonly used expectation-maximization (EM) algorithm, we would rather use a hierarchical agglomerative (bottom-up) k-means clustering (HAC). K-means clustering represents good trade-off between computational complexity and accuracy (by comparison with EM, covariance matrices are almost similar). The biggest advantage of HAC is, that the number of models c are not fixed to some value, but is adaptable with the different types of road surface (clusters are merged in the same way, as is discussed in the subsection Update). Each cluster c is represented by its mean vector $\boldsymbol{\mu}$, covariance matrix $\boldsymbol{\Sigma}$ and a mass which is equal to the number of pixels associated to each cluster

$$\boldsymbol{\mu}_c = \frac{1}{n_c} \sum_{i=1}^{i=n_c} \mathbf{p}_{c,i}, \tag{18}$$

$$\boldsymbol{\Sigma}_c = \frac{1}{n_c} \sum_{i=1}^{i=n_c} \mathbf{p}_{c,i}\mathbf{p}_{c,i}^{\mathrm{T}} - \boldsymbol{\mu}_c\boldsymbol{\mu}_c^{\mathrm{T}}, \tag{19}$$

$$mass_c = n_c \tag{20}$$

To ensure robustness of training models, all models which do not have at least $T_{outliers} = 15\%$ pixels of the most massive model are refused as outliers. To avoid troubles with uniformly colored roads, an explicit minimum noise term ϕI_{3x3} is added to the covariance matrix. Another posibillity of creating models, is usage of a fuzzy c-means clustering, which allows to be more/less selective whether the point belongs to the cluster or not, instead of standard k-means.

2) Update: In addition to c training models, n_l learned models exist, which represent "history of the road" with exponential forgetting. At the beginning, all color models are null. Each training model is compared with learned models

$$(\boldsymbol{\mu}_L - \boldsymbol{\mu}_T)^{\mathrm{T}}(\boldsymbol{\Sigma}_L + \boldsymbol{\Sigma}_T)^{-1}(\boldsymbol{\mu}_L - \boldsymbol{\mu}_T) \leq d_{similar}, \tag{21}$$

where $\boldsymbol{\mu}$ is a mean vector, and $\boldsymbol{\Sigma}$ is a covariance matrix. If the training model overlaps any learned model, the learned

model is updated according to formulas

$$\boldsymbol{\mu}_{updated} = \frac{m_L\boldsymbol{\mu}_L + m_T\boldsymbol{\mu}_T}{m_L + m_T}, \tag{22}$$

$$\boldsymbol{\Sigma}_{updated} = \frac{m_L\boldsymbol{\Sigma}_L + m_T\boldsymbol{\Sigma}_T}{m_L + m_T}, \tag{23}$$

$$m_{updated} = m_L + m_T, \tag{24}$$

where m is associated mass to the model. Otherwise, there are two possibilities. If all models are not full, then 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.

3) Shadows and overexposed highlights: Once the robot is among a shady and/or overexposed highlighted road segments, models with the same original color could be easily discarded after a few frames. It is the same situation when the robot moves away from these parts, however, more shady and/or highlighted road segments are straight forward. Thus, the models with high mass are compared with those with low mass. If the mean color of those models are similar, the mass of small models is adjusted to above some value (f_{shadow} multiplied by the mass of the most massive model). The comparison of mean colors is based on modified Hue proposed by Finlayson [19]. The models are similar, if both conditions are satisfied

$$|Hue(\boldsymbol{\mu}_i) - Hue(\boldsymbol{\mu}_j)| < H_T, \tag{25}$$

$$|Brightness(\boldsymbol{\mu}_i) - Brightness(\boldsymbol{\mu}_j)| > B_T, \tag{26}$$

$$Hue(\boldsymbol{\mu}_i) = \arctan\frac{\log r_i - \log g_i}{\log r_i + \log g_i - 2\log b_i}, \tag{27}$$

$$Brightness(\boldsymbol{\mu}_i) = \frac{r_i + b_i + g_i}{3}. \tag{28}$$

In addition to that, shadow and highlight "preprocessors" provides more information about the environment to higher AI [27]. It is important in situations when a robot is not yet among the shadowed/highlighted segments, however, these difficult illumination conditions are straight forward. Without preprocessors, a huge dark shadows, or overexposed highlights will be labeled as a non-road. Only pixels under line $\mathbf{l}$ determined by the vanishing point are considered. Both detectors are similar - intensity of each pixel is compared with some threshold and if the value is close enough to 0 for shadow or 1 for highlight preprocesor, pixel is masked

$$shadows(x,y) = \begin{cases} 1 & \text{if } intensity(\mathbf{p}) < T_{shadow}, \\ 0 & \text{otherwise,} \end{cases} \tag{29}$$

$$highlights(x,y) = \begin{cases} 1 & \text{if } intensity(\mathbf{p}) > T_{highlight}, \\ 0 & \text{otherwise,} \end{cases} \tag{30}$$

where $intensity(\mathbf{p}) = 0.299r + 0.587g + 0.114b$. Unfortunately, masked pixels do not contain enough information about color, thus, these pixels are not automatically labeled as road, however information about these regions are important for higher AI.

C. Adaptivity and Robustness

The mass of each model is an important value for road segmentation (discussed below). However, mass updating formula has an integral character. Consequently, this increases the robustness of the method, however it negatively influences speed of adaptivity. It is possible to solve this naively by a huge decay factor, which is taken off from mass at each frame, however this solution leads to the loss of models history (models do not remember more than last few frames). It is a similar task to the problem of anti-windup, which is well known from control theory of feedback systems. Good choice of appropriate limit is important, because it depends on the number of expected clusters produced by k-means (it expects the worst case - uniformly associated pixels to each cluster), adaptivity speed, which describes the worst case of how many frames it will take before the new model is used, and a factor $d_{classify}$ which is the worst case of threshold used for Mahalanobis scoring (discussed below). Thus, we add saturation nonlinearity with superior limit

$$AWU = \frac{n_{frames_1}}{d_{classify}}\frac{n_{tr}}{c}, \tag{31}$$

where n_{frames_1} is a number of frames which determines adaptivity speed, n_{tr} is size of a centered training area, c is an expected number of training models produced by HAC and $d_{classify}$ is a threshold for Mahalanobis distance measurement. Therefore we are able to set adaptivity speed without loss of models history.

On the other hand, we do not want to store models which were not updated for many frames. Hence, a decay factor from each learned model is taken off in each frame

$$D = AWU^{-\frac{1}{n_{frames_2}}}, \tag{32}$$

where n_{frames_2} is a number of frames for exponential forgetting.

D. Road Segmentation

Once all routines connected with management of models are done, 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 learned models. Only models with mass above some value $d_{classify}$ (fraction of the biggest model) are considered. The condition is important for both reasons - it improves the robustness of the method and saves computation time. The roadness score is measured as a minimum of Mahalanobis square norm

$$D(\mathbf{p}, \boldsymbol{\mu}_i) = \min_i((\mathbf{p} - \boldsymbol{\mu}_i)^{\mathrm{T}})\boldsymbol{\Sigma}_i^{-1}(\mathbf{p} - \boldsymbol{\mu}_i)) \tag{33}$$

Next, it is possible to use these values as an input of probabilities to some higher AI (e.g. occupancy grids, ...), or identify patches that create the road. To extract only road segments, we run thresholding with an adaptive threshold. The default threshold is determined by pixels belonging to the training area (pixels labeled as outliers by k-means are excluded) - the threshold is set to $\mu + 3\beta\sigma$, which ensure that all pixels in the training area are selected as road pixels. Nevertheless, we expect that at least 25% of image

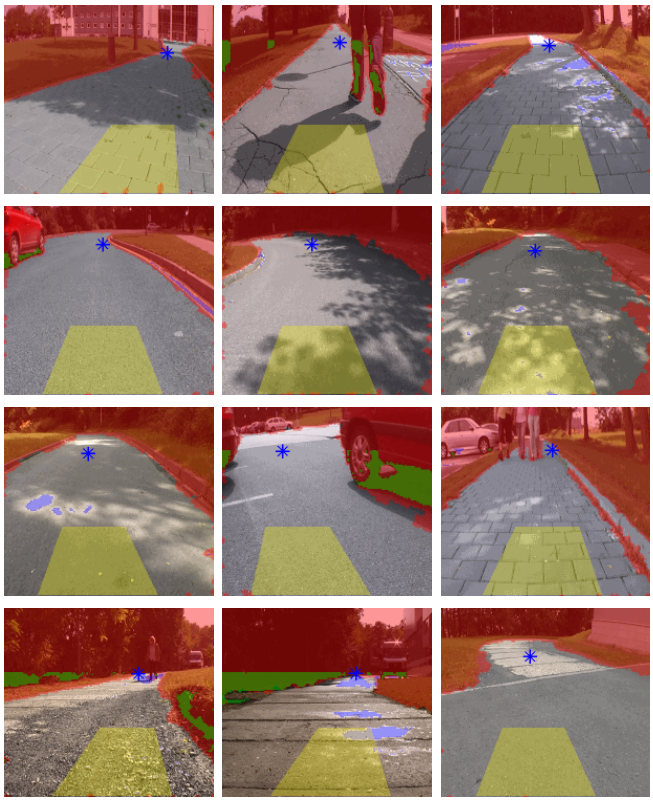


Fig. 6. Fusion of frequency based vanishing point estimation and probabilistically based texture segmentation - performance against various road types, illumination and obstacles. Blue star is the estimated vanishing point, yellow trapezoid is a training area, the blue area denotes highlight preprocessor and green is the shadow preprocessor.

is created by non-road pixels. Thus, once the thresholding is done, the non-road pixels are summed up. If the number of non-road pixels is below 25%, parameter β decreases and thresholding start again. In fact, it usually is $\beta = 1$, however if the $c_1c_2c_3$ color model fails, β descreases to ensure correct classification, but these situations are rare. To remove small areas labeled as non-road and preserve large obstacles, morphological operations dilation an erosion are performed. Finally, only that blob, which is connected with the training area by flood fill, is preserve as a road region, others are discarded as non-road.

VI. LESSONS LEARNED

The proposed algorithm was tested on a number of different sequences, which consist from more than 10 000 images captured by Orpheus-AC[1]. In contrast to e.g. domain of local invariant feature detectors and descriptors, no standardized dataset and performance evaluation framework exists, like for features does [28], [29]. Moreover, the test sequences of previously published papers [10], [12] are not freely available.

Currently, our algorithm is implemented in Matlab, however, a subsequent report, focused on the efficient implementation for DSP and FPGA will follow in near future. The solution will be based on a Sundance SMT372T platform, which is powerful enough to provide a real-time response, moreover it is available in a military-grade version. The input image was subsampled to $(I_w, I_h) = (128, 128)$ which is the best trade-off between computational complexity and accuracy. All results were obtained with the following parameters: $\lambda = 5$, $k = 16$, $C_{top} = \frac{1}{4}I_h$, $n_l = 15$, $h = \frac{4}{5}I_h$, $T_{outliers} = 15\%$, $d_{similar} = 1$, $H_T = 0.2$, $B_T = 30$, $T_{shadow} = 0.1$, $T_{highlight} = 0.85$, $f_{shadow} = 0.2$, $n_{frames_1} = 5$, $n_{frames_2} = 200$ and $c_1c_2c_3$ color space.

Our algorithm performs well in various environments consisting of different types of pavements, ill-structured rural and (sub)urban roads. The algorithm is robust enough to work in different illumination conditions, including dark cast shadows and overexposed highlights. Shadows and white preprocessors label image areas that do not contain enough information about color. Such information can be used by a higher AI system. Moreover, we are able to specify the adaptivity speed and quantity of models stored in a history archive. Finally, due to the fusion of frequency and probabilistically based approaches, the algorithm is robust against sudden changes of road surfaces.

The vanishing point estimation method was originally proposed for desert roads [14]. However, our experiments showed, that for a (sub)urban environment, the following conditions should be fulfilled: when there are no significant dominant orientations in the texture, it is necessary to use wide lenses to ensure that enough of both road borders will be in the image. Further, we changed the image subsampling rate, sizes of Gabor kernels and their associated wavelength λ and a parameter σ. The estimation of dominant orientations is performed in a B channel of RGB color space to suppress different illumination conditions. In addition to that, some dominant orientations $\theta_{max}(\mathbf{p})$ are rejected from voting and it is convenient to use the particle filter to avoid the misestimation of a vanishing point.

By comparison with state-of-the-art methods [10], [12], our training area is defined without any estimation of a 3D depth map. Thus, we are able to distinguish e.g. pavements and other areas like grass, etc. without any high borders. Consequently, our method does not use the whole ground plane, however, we are able to select a drivable path with higher precision. On the other hand, due to the HAC, we can remove outliers (obstacles, color noise, ...) which differ in either color or height. Sliding of a training area, is useful when the robot is close enough to the borders of a path to avoid learning of non-road colors.

An anti-windup and decay factor are complementary coefficients dealing with better management of the previously learned models in the history archive (see Fig. 7). The importance of a sliding training area is shown on Fig. 8.

VII. CONCLUSIONS AND PERSPECTIVES

We have presented a robust approach to the monocular camera based extraction of shady and highlighted roads for UGV. Our approach does not need any additional sensor or difficult calibration. It works well on both unstructured and (semi)structured roads, with various types of surfaces

[1] Videos can be found at http://www.miksik.co.uk.

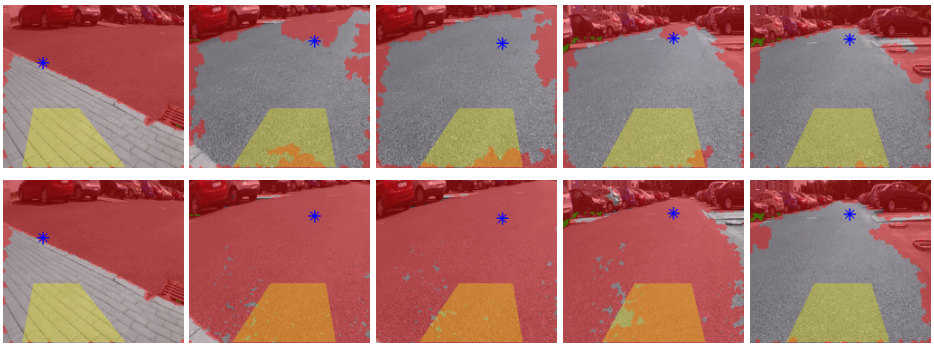


Fig. 7. Comparison of road extraction with properly set anti-windup and decay factor (top row) and processing without these factors (bottom row). In both cases (even if texture segmentation fails), it is still possible to successfully navigate the robot, because the vanishing point can be used.

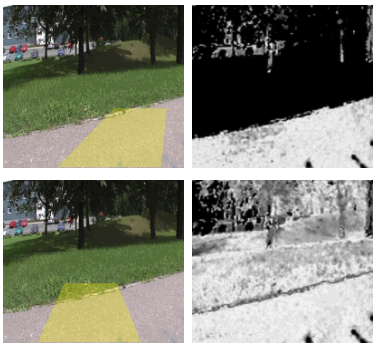


Fig. 8. Comparison of drivability maps produced with a sliding training area (top row) and fixed training area (bottom row).

and dynamically changing light conditions, including dark cast shadows and overexposed highlights. Due to the novel fusion of a frequency based vanishing point estimation and a probabilistically based texture segmentation, it can be used even in cases when the road borders are not hight, which is the limitation of previous approaches. Dynamic properties can be controlled by complementary anti-windup and decay factors. Besides, a fusion of two different approaches leads to better robustness, because even if one of them fails, it is still possible to successfully navigate the robot.

A subsequent report, focused on the implementation of our algorithm into the embedded system will follow in the near future.

REFERENCES

[1] L. Zalud, *Robocup 2003: Robot Soccer World Cup VII*. Springer-Verlag, 2004, ch. Rescue Robot League - 1st Place Award Winner.

[2] ——, "Orpheus - reconnaissance teleoperated robotic system," in *16th IFAC World Congress, Prague, Czech Republic*, 2005.

[3] J. Crisman and C. Thorpe, "Unscarf, a color vision system for the detection of unstructured roads," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 3, April 1991, pp. 2496 – 2501.

[4] ——, "Scarf: A color vision system that tracks roads and intersections," *IEEE Trans. on Robotics and Automation*, vol. 9, no. 1, pp. 49 – 58, February 1993.

[5] A. Broggi, M. Bertozzi, and A. Fascioli, "Argo and the millemiglia in automatico tour," *IEEE Intelligent Systems*, vol. 14, no. 1, pp. 55–64, 1999.

[6] M. Bertozzi, A. Broggi, A. Fascioli, and S. Nichele, "Stereo vision-based vehicle detection," in *in IEEE Intelligent Vehicles Symposium*, 2000, pp. 39–44.

[7] D. Pomerleau, "Neural network vision for robot driving," in *The Handbook of Brain Theory and Neural Networks*, M. Arbib, Ed., 1995.

[8] D. Lieb, A. Lookingbill, and S. Thrun, "Adaptive road following using self-supervised learning and reverse optical flow," in *Robotics: Science and Systems*, S. Thrun, G. S. Sukhatme, and S. Schaal, Eds. The MIT Press, 2005, pp. 273–280.

[9] A. Wedel, T. Pock, C. Zach, H. Bischof, and D. Cremers, "An improved algorithm for tv-l1 optical flow," *Statistical and Geometrical Approaches to Visual Motion Analysis: International Dagstuhl Seminar*, pp. 23–45, 2009.

[10] H. Dahlkamp, A. Kaehler, D. Stavens, S. Thrun, and G. R. Bradski, "Self-supervised monocular road detection in desert terrain," in *Robotics: Science and Systems*, G. S. Sukhatme, S. Schaal, W. Burgard, and D. Fox, Eds. The MIT Press, 2006.

[11] A. Broggi, A. Fascioli, A. Member, and R. Fascioli, "Artificial vision in extreme environments for snowcat tracks detection," *IEEE Trans. on Intelligent Transportation Systems*, vol. 3, pp. 162–172, 2002.

[12] T.-C. Dong-Si, D. Guo, C. H. Yan, and S. H. Ong, "Robust extraction of shady roads for vision-based ugv navigation," in *IROS*. IEEE, 2008, pp. 3140–3145.

[13] B. Ma, S. Lakshmanan, and A. O. Hero, "Simultaneous detection of lane and pavement boundaries using model-based multisensor fusion," *IEEE Transactions on Intelligent Transportation Systems*, 2000.

[14] C. Rasmussen, "Grouping dominant orientations for ill-structured road following," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2004.

[15] C. Rasmussen and T. Korah, "On-vehicle and aerial texture analysis for vision-based desert road following," in *IEEE International Workshop on Machine Vision for Intelligent Vehicles*, 2005.

[16] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, "Evaluation of color descriptors for object and scene recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, Alaska, USA, 2008.

[17] G. D. Finlayson, S. D. Hordley, M. S. Drew, and E. N. Tj, "Removing shadows from images," in *In ECCV 2002: European Conference on Computer Vision*, 2002, pp. 823–836.

[18] Z. Figov, Y. Tal, and M. Koppel, "Detecting and removing shadows," in *Proc. of 7th IASTED Int'l Conference on Computer Graphics and Imaging*, Kaui HW, August 2004.

[19] G. D. Finlayson and G. Schaefer, "Hue that is invariant to brightness and gamma," in *BMVC*, T. F. Cootes and C. J. Taylor, Eds. British Machine Vision Association, 2001.

[20] R. Ghurchian and S. Hashino, "Shadow compensation in color images for unstructured road segmentation," in *MVA*, 2005, pp. 598–601.

[21] J. P. Jones and L. A. Palmer, "An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex," *J Neurophysiol*, vol. 58, no. 6, pp. 1233–1258, December 1987.

[22] T. S. Lee, "Image representation using 2D gabor wavelets," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, pp. 959–971, 1996.

[23] M. Hansen and G. Sommer, "Active depth estimation with gaze and vergence control using gabor filters," in *In In 13th Int. Conf. on Pattern Recognition, Volume A*, 1996, pp. 287–291.

[24] M. Frigo and S. G. Johnson, "The design and implementation of FFTW3," *Proceedings of the IEEE*, vol. 93, no. 2, pp. 216–231, 2005, special issue on "Program Generation, Optimization, and Platform Adaptation".

[25] S. M. Ettinger, M. C. Nechyba, P. G. Ifju, and M. Waszak, "Vision-guided flight stability and control for micro air vehicles," *Advanced Robotics*, vol. 17, no. 7, pp. 617–640, 2003.

[26] M. Isard and A. Blake, "Condensation - conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.

[27] K. Berger, C. Lipski, C. Linz, T. Stich, and M. Magnor, "The area processing unit of caroline - finding the way through darpas urban challenge. robvis," 2008.

[28] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, 2005.

[29] O. Miksik, "Fast feature matching for simultaneous localization and mapping," Bachelor's Thesis, Brno University of Technology, 2010.