Direct Superpixel Labeling for Mobile Robot Navigation Using Learned General Optical Flow Templates

Richard Roberts[1] and Frank Dellaert[1]

***Abstract*— Towards the goal of autonomous obstacle avoidance for mobile robots, we present a method for superpixel labeling using optical flow templates. Optical flow provides a rich source of information that complements image appearance and point clouds in determining traversability. While much past work uses optical flow towards traversability in a heuristic manner, the method we present here instead classifies flow according to several optical flow templates that are specific to the typical environment shape. Our first contribution over prior work in superpixel labeling using optical flow templates is large improvements in accuracy and efficiency by inference directly from spatiotemporal gradients instead of from independently-computed optical flow, and from improved optical flow modeling for obstacles. Our second contribution over the same is extending superpixel labeling methods to arbitrary camera optics without the need to calibrate the camera, by developing and demonstrating a method for learning optical flow templates from unlabeled video. Our experiments demonstrate successful obstacle detection in an outdoor mobile robot dataset.**

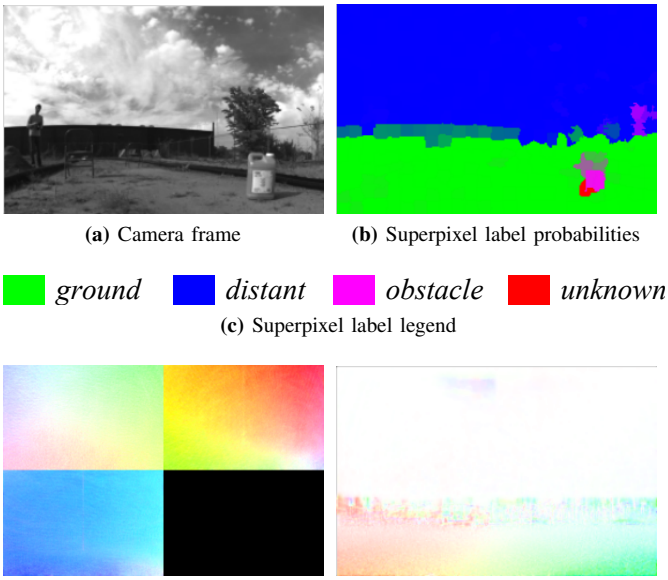
I. Introduction

We are interested in autonomous obstacle avoidance for mobile robots using camera sensors. Most of the work to date on this problem has focused primarily on either building and analyzing sparse point clouds, or labeling pixels or superpixels based on image appearance. However, both these families of methods have shortcomings, and optical flow provides an additional powerful source of information to complement point clouds and image appearance.

The most ubiquitous methods to date in autonomous robot systems using camera input have focused on two families of methods. In one, stereo or structure-from-motion is used to build point clouds or depth maps, which are analyzed to estimate a ground traversability map, for example [14], [16], [8]. In [7], [11], and [10], image appearance traversability classifiers, combined with a model of the ground plane, are used to propagate stereo-labeled image regions to distances in the traversability map beyond the range of stereo. Many successful methods also use image appearance to propagate traversability but without the use of nearby stereo-informed labels [13], and others use examples provided by a human operator to inform labeling [21]. In most of the working systems described by the preceding papers, stereo and appearance traversability are combined to produce the traversability map used for motion planning. We review additional related work in Section II.

*This work is supported in part by MURI grant no. W911NF1110046.

[1]The authors are with the Institute for Robotics and Intelligent Machines, Georgia Institute of Technology, Atlanta, GA 30332, USA. Corresponding author: richard.jw.roberts@gmail.com.



(a) Camera frame **(b)** Superpixel label probabilities

(c) Superpixel label legend

(d) Learned rotational optical flow template **(e)** Learned translational optical flow template

Fig. 1: This paper is concerned with obstacle avoidance for mobile robots *a)* from an uncalibrated camera sensor with arbitrary optics. Here, the large radial distortion which cannot be undistorted by typical methods. *b)* Towards this goal our method labels superpixels via image motion directly from spatiotemporal image gradients. *d,e)* Our method learns *optical flow templates* to inform this labeling, with no need to hand-label training data.

The above methods are ill-suited for small or inexpensive robots. The point clouds and depth maps of the first family of methods must be fairly dense to support obstacle avoidance, which necessitates expensive and powerful hardware. Additionally, they require a calibrated camera, and calibrating wide-angle lenses, which are certainly useful for obstacle avoidance, is challenging and requires special models and methods. While appearance-based classification methods are computationally fast and do not rely on calibration, they either must be adapted online using information from point clouds or depth maps, or trained offline and thus unable to cope with unexpected appearance changes.

Optical flow provides a rich source of information to complement image appearance and point clouds in determining traversability. To demonstrate this, we present a method for labeling superpixels in video as *ground*, *distant*, *obstacle*, or *unknown* using as input the observed spatiotemporal image gradients, and a learned optical flow template model, as previewed in Figure 1. We additionally present a method for learning these optical flow templates.

The contributions of this paper over previous work in superpixel labeling with optical flow templates [17] are that we *1)* improve labeling accuracy and computational efficiency over prior work by inferring labels *directly from spatiotemporal gradients*, instead of from separately-computed optical flow, *2)* improve labeling accuracy by improved modeling of the *obstacle* class, and *3)* expand applicability to uncalibrated cameras of arbitrary optics, by developing a method for unsupervised learning of optical flow templates from video.

II. Related Work

Recent mobile robot scene understanding work leverages image appearance and 3D information from structure from motion. Brostow *et al*. [1] and Sturgess *et al*. [19] use such information to semantically label video as street, sidewalk, and car. Geiger *et al*. [4] and Zhang *et al*. [22] go beyond pixel labels, estimating the 3D shape of intersections and traffic flow, leveraging the output of algorithms for vehicle tracking, appearance-based pixel labeling, other tasks.

Optical flow for autonomous navigation has been the focus of much work. Giachetti *et al*. [5] labels cars using the consistency of observed optical flow with that predicted under a ground plane assumption. Nourani-Vatani *et al*. [15] use optical flow to recognize changes in coarse environment shape using spatial statistics of observed optical flow fields, and compare them against a recorded database.

Classic work in navigation from optical flow has used heuristics to control robots directly from measured optical flow. This work has its base in animal and human studies of the use of optical flow in navigation [6], [12]. Bio-inspired control laws have been developed for pursuit, escape, hallway following, and obstacle avoidance, for example Duchon *et al*. [3] and as reviewed in Srinivasan *et al*. [18].

More recent work in optical flow navigation has used more informed optical flow patterns beyond heuristics. Conroy *et al*. [2] and Hyslop and Humbert [9] infer similarity of the observed optical flow field to that expected for coarse environment structures such as walls and corridors from a set of template optical flow fields, using "wide-field optic flow integration". They also develop control laws to apply this to autonomous robot navigation.

III. Methods

Our goal is to perform online labeling of superpixels according to the scene structure at each superpixel, using as input the spatiotemporal image gradients. To inform this labeling, we use learned models of the optical flow fields due to platform egomotion for a set of possible typical scene structure, called *optical flow templates*. Note that we perform this labeling *without* computing optical flow – dense optical flow calculation is ill-posed and unnecessarily expensive, which is why we instead directly use the spatiotemporal image gradients. Additionally, no velocity information is needed to perform the labeling.

Before online labeling, there is an offline phase to learn these optical flow templates. No velocity estimates, camera calibration, or hand-labeling are required for learning – it is almost entirely unsupervised.

We will explain our method in three subsections. First, in III-A we introduce the optical flow template model, which drives the labeling. Second, in III-B we explain how we use learned optical flow templates to label superpixels. Finally, in III-C we explain how we learn these templates.

A. Optical Flow Templates

An *optical flow template* models the optical flow at each pixel resulting from typical environment structure, for fixed but arbitrary camera optics, for any possible platform velocity. This model is generative and defines the likelihood $p\left(U_j \mid \Lambda_i, \omega, v, \Theta\right)$ of the flow U_j at each pixel j, given the platform rotational ω and translational v velocities, the the label Λ_i of the superpixel i that contains pixel j (the labeling may be done at any granularity but in this paper we do it over superpixels for reduced computation), and a set of optical flow template parameters Θ.

As mentioned, optical flow templates implicitly encode typical scene structure, and the label Λ_i determines which typical environment structure predicts the optical flow U_j for a pixel. Here, Λ_i=*ground* predicts optical flow consistent with a ground plane, Λ_i=*distant* corresponds to distant structure where flow is determined only by camera rotation, Λ_i=*obstacle* corresponds to structure nearer to the camera than that expected for the ground plane, and Λ_i=*unknown* predicts zero-mean wide-variance optical flow to identify superpixels that cannot be explained by the learned templates.

Mathematically, the optical flow template model predicts the optical flow U_j as

$$U_j = \begin{cases} \epsilon_{\textit{unknown}}, & \Lambda_i\text{=}\textit{unknown} \\ W_{\omega,j}\omega + W_{\Lambda_i,j}v + \epsilon_{\Lambda_i}, & \Lambda_i\text{=}\textit{ground}, \textit{obstacle} \\ W_{\omega,j}\omega + \epsilon_{\textit{distant}}, & \Lambda_i\text{=}\textit{distant}, \end{cases} \tag{1}$$

where $\omega \in \mathbb{R}^3$ is the platform rotational velocity, $v \in \mathbb{R}^q$ is the platform translational velocity, and ϵ_x denotes Gaussian noise with covariance Σ_x (with 'x' representing one of the labeling classes possible in Λ_i).

The W matrices encode the linear relationship between platform velocity and optical flow for each template. $W_{\omega,j} \in \mathbb{R}^{2\times 3}$ models the optical flow due to platform rotation, and $W_{\Lambda_i} \in \mathbb{R}^{2\times q}$ generates the optical flow due to platform translation for a particular environment structure.

For compactness we assign an index to each possible label. We assign $\Lambda_i = \textit{unknown} = 0$, $\Lambda_i = \textit{distant} = \kappa$, and otherwise $0 < \Lambda_i < \kappa$. q is the dimensionality of translational velocity accounting for non-holonomic constraints. Thus for the car-steering robot used in our experiments, $q = 1$. The full set of parameters is thus $\Theta = (W_\omega, W_1, \Sigma_1, \ldots, W_\kappa, \Sigma_\kappa)$.

Two intuitive points regarding optical flow templates are:

- Optical flow due to rotation is independent of the scene structure, the contribution due to platform translation is the only one that informs the pixel labels.
- For any *constant* environment structure relative to the robot, the optical flow is indeed linear in the platform

translational velocity v as modeled in Eq. 1. Optical flow is also linear in the platform rotational velocity ω. The above explanation is self-contained, but [17] may be referenced for additional information.

B. Labeling Superpixels Using Optical Flow Templates

We wish to infer superpixel labels Λ_i for each i^{th} superpixel in an image, given the image spatiotemporal gradients $I^{txy} = (I^t, I^x, I^y)$ and the learned template parameters Θ. We will perform inference in the generative model illustrated in Figure 2, whose joint density is

$$p\left(I^{txy}, U, \Lambda, \omega, v \mid \Theta\right) = p\left(I^{txy} \mid U\right) p\left(U \mid \Lambda, \omega, v, \Theta\right) p\left(\Lambda\right) p\left(\omega, v\right). \quad (2)$$

The unknowns are the labels $\Lambda = \{\Lambda_i\}$ for all superpixels, the robot velocity ω, v, and the optical flow $U = \{U_j\}$ for all pixels, while the observations are the image spatiotemporal gradients I^{txy}.

Exact inference of the superpixel labels would require an intractable marginalization of the unknown flow U and velocity ω, v, so we instead employ variational inference. We alternate between updating the label Λ probabilities, and a Gaussian approximation of the robot velocity. The inference problem is thus approximated as

$$p\left(\Lambda \mid I^{txy}, \Theta\right) \approx \int_{\omega,v} p\left(\Lambda \mid I^{txy}, \omega, v, \Theta\right) \mathcal{N}\left(\left(\hat{\omega}, \hat{v}\right), \hat{\Sigma}\right), \quad (3)$$

where $\hat{\omega}, \hat{v}$ is the mean and $\hat{\Sigma}$ is the covariance of a Gaussian density estimate that we will iteratively update. We will derive a simple closed form expression for Eq. 3 at each iteration. The two steps in each iteration are:

a) Label Probability Update: A vector of label probabilities $\tilde{\Lambda}_i \in \mathbb{R}^{\kappa+1}$ for each superpixel is updated by evaluating Eq. 3. The first term in Eq. 3 is obtained by applying Bayes' law and marginalizing out the unknown optical flow U,

$$\begin{aligned} p\left(\Lambda \mid I^{txy}, \omega, v, \Theta\right) &= \prod_i p\left(\Lambda_i \mid I^{txy}_{J_i}, \omega, v, \Theta\right) \\ &\propto \prod_i \prod_{j \in J_i} \int_{U_j} p\left(I^{txy}_j \mid U_j\right) p\left(U_j \mid \Lambda_i, \omega, v, \Theta\right) p\left(\Lambda_i \mid \Theta\right), \end{aligned} \quad (4)$$

where J_i is the set of pixels in superpixel i. Note that in the first line, the superpixel labels are independent of each other when conditioned on the robot velocity ω, v, and the second line is obtained by applying Bayes' law multiple times.

The first term $p\left(I^{txy}_j \mid U_j\right)$ in Eq. 4 is the image intensity likelihood which, assuming a static and non-occluding scene, depends only on the optical flow. Thus we define it using the brightness constancy constraint,

$$p\left(I^{txy}_j \mid U_j\right) = \mathcal{N}\left(0;\, I^t_j + I^{xy}_j U_j, \sigma_I\right), \quad (5)$$

where $I^{xy}_j \in \mathbb{R}^{1\times 2}$ is the spatial image gradient at pixel j, and σ_I is the standard deviation of pixel intensity noise. The second term $p\left(U_j \mid \Lambda_i, \omega, v, \Theta\right)$ is the optical flow likelihood according to the optical flow template model in Eq. 1.

The label prior $p\left(\Lambda_i \mid \Theta\right)$ is a categorical distribution. The outlier class *unknown* is assigned a small probability ($p\left(\Lambda_j{=}\textit{unknown}\right) = 0.05$ in our experiments). Then, the region of pixels close to the horizon receives equal probability *distant* and *ground*, and slighly lower probability *obstacle*. The pixels above that region receive zero *ground* probability, and the pixels below that region receive zero *distant* probability.

Since all terms in Eq. 4 are Gaussian as just explained, the integral in Eq. 4 can be computed analytically as

$$p\left(\Lambda \mid I^{txy}, \omega, v, \Theta\right) = \mathcal{N}\left(I^t_j;\, -I^{xy}_j \left(W_{\omega,j}\hat{\omega} + W_{\Lambda_i,j}\hat{v}\right), \rho_j\right) p\left(\Lambda_i \mid \Theta\right) \quad (6)$$

with

$$\rho_j = \sigma_i^2 + I^{xy}_j \left(\Sigma_{\Lambda_j} + W_{\omega\Lambda_j,j}\hat{\Sigma}_{\omega v} W^{\mathsf{T}}_{\omega\Lambda_j,j}\right) I^{xy\,\mathsf{T}}_j,$$

where $W_{\omega\Lambda_j,j} = [W_{\omega,j} \;\; W_{\Lambda_j,j}]$, and $\hat{\omega}, \hat{v}$ and $\hat{\Sigma}_{\omega v}$ are the statistics of the Gaussian density estimate on velocity from the previous iteration, whose calculation we will now explain.

b) Velocity Gaussian Density Update: As mentioned, we update the mean $\hat{\omega}, \hat{v}$ and covariance $\hat{\Sigma}_{\omega v}$ of the estimated velocity density using the label probability vectors $\tilde{\Lambda}_i$ from the previous iteration, using the same generative model,

$$\begin{aligned} \hat{\omega}, \hat{v} &\leftarrow \arg\max_{\omega,v} \left\langle \log p\left(\omega, v \mid I^{txy}, \Lambda, \Theta\right)\right\rangle \\ &= \arg\max_{\omega,v} \sum_i \sum_{j\in J_i} \left\langle \log p\left(I^{txy}_j \mid \Lambda_i, \omega, v, \Theta\right)\right\rangle + \log p\left(\omega, v\right) \\ &= \arg\max_{\omega,v} \sum_i \sum_{j\in J_i} \sum_{k\in\{0..\kappa\}} \tilde{\Lambda}_{i,k} \log p\left(I^{txy}_j \mid \Lambda_i, \omega, v, \Theta\right) \\ &\qquad + \log p\left(\omega, v\right) \end{aligned} \quad (7)$$

where the prior $p\left(\omega, v\right)$ is assumed to be uninformative and

$$p\left(I^{txy}_j \mid \Lambda_i, \omega, v, \Theta\right) = \mathcal{N}\left(I^t_j;\, -I^{xy}_j \left(W_{\omega,j}\omega + W_{\Lambda_i,j}v\right), \eta_j\right) \quad (8)$$

where $\eta_j = \sigma_I^2 + I^{xy}_j \Sigma_{\Lambda_i} I^{xy\,\mathsf{T}}_j$ is obtained by again integrating out the unknown flow U. The maximization in Eq. 7 is a linear least-squares problem.

The covariance estimate $\hat{\Sigma}_{\omega v}$ is that of the Gaussian defined by the exponential of the expected log-likelihood in Eq. 7, $\exp\left\langle \log p\left(\omega, v \mid I^{txy}, \Lambda, \Theta\right)\right\rangle$, and is obtained efficiently as $\hat{\Sigma}_{\omega v} = \left(R^{\mathsf{T}} R\right)^{-1}$, where $R \in \mathbb{R}^{(3+q)\times(3+q)}$ is the Cholesky factor obtained in optimizing Eq. 7.

C. Learning Optical Flow Templates from Unlabeled Video

We wish to learn the optical flow templates to label superpixels in Section III-B from recorded video. As mentioned, the learning algorithm is almost entirely unsupervised.

The input to the learning algorithm is two videos – one while arbitrarily rotating the camera (hand-held is sufficient) in various directions, and one while the robot drives safely in its typical environment.

The learning algorithm is also a variational method that sequentially updates the probability vectors of the pixel labels $\tilde{\Lambda}_{t,j}$ for every *pixel* (not superpixel) j in frame t, velocity estimates ω_t, v_t for every frame, and the optical flow templates W_ω and W_k for each template k.

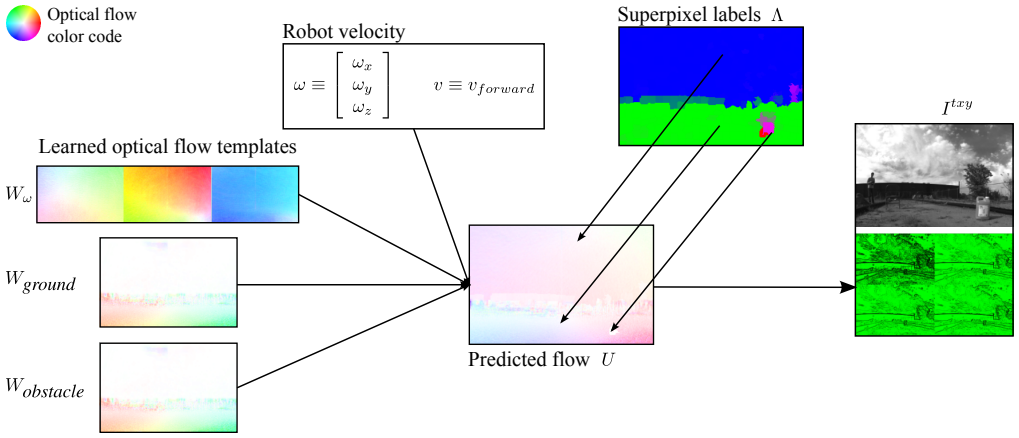


Fig. 2: Illustrated Bayes net of the optical flow template model.

The optical flow templates are updated by maximizing their expected log-likelihood in the same generative model as above, conditioning on estimates of the other unknowns from the previous iteration. Unlike in the inference algorithm, in learning it suffices to condition on fixed values for ω, v instead of using the Gaussian density,

$$W_{\omega,1..\kappa} = \arg\max_{W_{\omega,1..\kappa}} \left\langle \log p\left(W_{\omega,1..\kappa} \mid I^{txy}_{1..T}, (\Lambda, \hat{\omega}, \hat{v})_{1..T}\right)\right\rangle$$
$$= \arg\max_{W_{\omega,1..\kappa}} \left\langle \sum_t \log p\left(I^{txy}_t | (\Lambda, \hat{\omega}, \hat{v})_t, \Theta\right)\right\rangle + \log p\left(W_{\omega,1..\kappa}\right), \tag{9}$$

where $W_{\omega,1..\kappa}$ is all optical flow templates W_ω and W_k for $k \in \{1..\kappa\}$, and T is the number of frames in the training dataset. The spatiotemporal gradient likelihood term $p\left(I^{txy}_t | (\Lambda, \hat{\omega}, \hat{v})_t, \Theta\right)$ is identical to Eq. 8, just for frame t.

The updates for the label probabilities $\tilde{\Lambda}_{t,j}$ and velocity mean $\hat{\omega}_t, \hat{v}_t$ are identical to those for the inference algorithm in Eqs. 4 and 7 (the notation of those update equations may then be interpreted with each superpixel containing only one pixel each), except that the priors for learning that appear in those equations are defined as follows:

- The label priors $p\left(\Lambda_{t,j} \mid \Theta\right)$ are almost the same as during inference in Section III-B, except that for the frames that are part of the rotation-only video *ground* receives zero probability, and the *obstacle* class probability is always zero for learning.
- The velocity prior $p(\omega, v)$ is Gaussian with covariance $\mathrm{diag}\left(\sigma^2_{\omega v}\right)$, $\sigma_{\omega v} = 10^4$. This merely constrains the optical flow scale ambiguity.
- $p\left(W_{\omega,1..\kappa}\right)$ is a gentle 4-connected smoothness prior on the optical flow templates:

$$p\left(W_{\omega,1..\kappa}\right) = \prod_{x \in \{\omega,1..\kappa\}} \prod_j \prod_{n \in N(j)} \mathcal{N}\left(W_{x,j} - W_{x,n}, \Sigma_b\right),$$

  where $N(j)$ is the two neighboring pixels of j to the right and below, and $\Sigma_b = \mathrm{diag}\left(\sigma^2_b\right)$, with $\sigma_b = 2$.

The learning is not highly sensitive to the parameters, and the ones we used here should also be appropriate for most other datasets. See table I for a summary of the parameters.

Over iterations, W_ω converges to the rotational flow fields and W_{ground} to the translational flow fields. Intuitively, the following points are responsible for this convergence:

- For each label class, i.e. typical structure, the templates identify a subspace $[\, W_\omega \, W_\Lambda \,]$ in the training data, converging for the same reason as in [20].
- The separation of rotational and translational flow fields is a model selection phenomenon – rotational flow fields contribute to all label classes while translational flow fields contribute to only non-*distant* classes, so the Gaussian prediction of optical flow U_j has smaller variance and thus higher probability if the rotation-only *distant* model is chosen where possible.

Instead of directly estimating a $W_{obstacle}$ template for the obstacle class, we leverage the fact that the contribution of translational flow of static obstacles is in the same *direction* as translational flow for any other structure (and still includes the same rotational flow component as any other structure). Obstacles are closer to the robot than the ground plane, yielding larger translational flow. Thus, we set $W_{obstacle} = \alpha W_{ground}$, with $\alpha = 1.1$.

Note that the full system Jacobians of the learning updates may be too large to fit in memory. Thus, during each learning iteration we accumulate the system Hessian via updates with each successive frame. Each learning iteration thus requires one pass through the data on disk.

IV. Experiments and Results

A. Sensitivity to Parameters

Most of the parameters in our method do not affect the results much, except that the inference algorithm is fairly sensitive to the optical flow covariance Σ_k and pixel intensity standard deviation σ_I. We tuned these empirically by visually inspecting our method's results.

The set of parameters listed in Table I are typically usable as-is on any other dataset, though the flow covariance Σ_k must be scaled according to resolution and field-of-view.

B. Experimental Platform and Setup

We collected learning and evaluation videos from the robot in Figure 3. Videos were collected by driving the robot

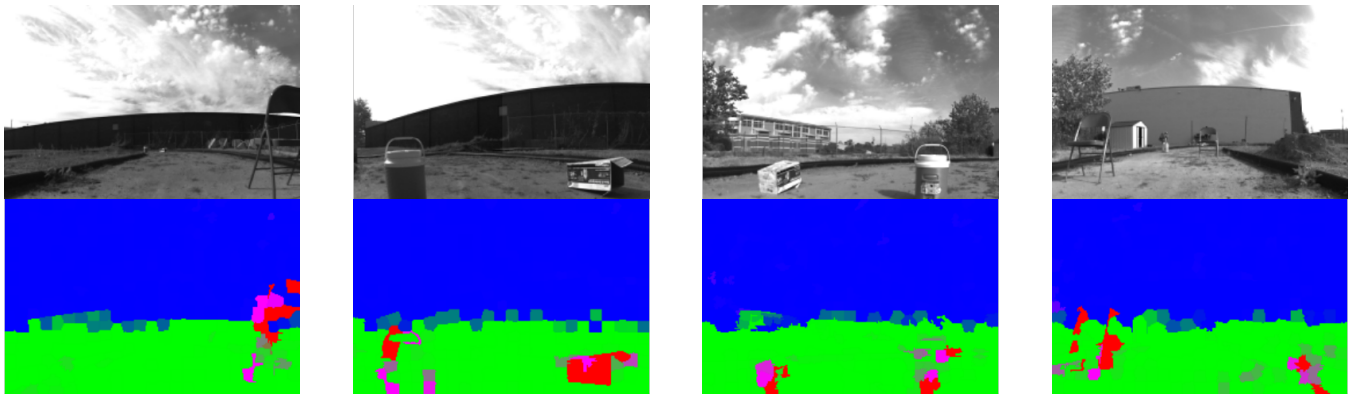


Fig. 4: Successful labeling results, see Section IV-C for observations and explanations.

Fig. 3: Our platform is a high-speed mobile robot, a modified $1/8$-scale radio-controlled car approximately $50\,\mathrm{cm}$ in length with on-board sensors and computing. The camera has a high-distortion 110° FOV lens. The robot was operated at approximately $2\mathrm{m/s}$ during data collection.

Description	Symbol	Value
Prior for *unknown* class	$p\left(\Lambda_{t,j}{=}\textit{unknown}\right)$	0.05
Pixel intensity noise (img. intensities in $[0,1]$)	σ_I	0.02
Optical flow prediction noise for *unknown* class	$\Sigma_{\textit{unknown}}$	$\mathrm{diag}\left(1.0\,\mathrm{pix}\right)^2$
Optical flow prediction noise for other classes	$\Sigma_{\omega,1..\kappa}$	$\mathrm{diag}\left(0.35\,\mathrm{pix}\right)^2$
Learning velocity prior	$\sigma_{\omega v}$	10^4
Learning template smoothness	σ_b	2
Obstacle template scaling	α	1.1
Superpixel average size		$100\,\mathrm{pix}^2$
Iterations for inference		3

TABLE I: Parameters selected for our experiments.

manually around an oval test track. In *training* videos the track was mostly free of obstacles, though tufts of grass and nearby structures were outliers for the learning algorithm. In the *evaluation* videos, we placed obstacles on the track.

C. Qualitative Results

The optical flow templates learned by our method are shown in Figures 1d and 1e. As described, linear combinations of the three rotational flow fields yield the contribution to optical flow from rotational velocity. The translational template is a single flow field because this robot has only one degree of freedom for body frame "forward velocity".

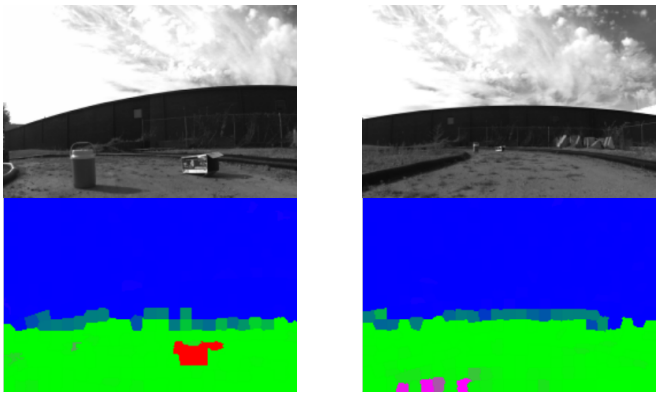


Fig. 5: Two types of failure for our method, see Section IV-C for observations and explanations.

Successful labeling examples are shown in Figure 4. Typically, boundaries of objects are detected more strongly than the smooth regions within objects because smooth texture provides weak information to optical flow. These results also show that *ground* and *distant* structure are well-distinguished, with distant structure often being sky.

Obstacles are sometimes classified as *unknown*. This is because the *obstacle* template, being a scaling in magnitude of the *ground* template, is exact only for obstacles at a particular depth. The *unknown* class thus captures regions whose flow is neither explained by the *ground* model nor the particular *obstacle* model. Improving the modeling of the *obstacle* template is part of our future work, but presently we note that the union of the *obstacle* and *unknown* labels is a good indicator of the presence of obstacles.

Most obstacles are detected several meters from the robot. Large obstacles are typically detected up to 10m and small or thin obstacles as close as 2m away. Optimizing and evaluating the detection range is a topic of our future work.

Figure 5 shows the types of failure cases. First, small objects directly at the focus-of-expansion of the image motion cannot be detected because image motion there is near zero. Second, sometimes a few superpixels of false positives appear at the bottom of the image when the robot speed is very large relative to the camera frame-rate, and the large motion between frames degrades the image spatio-temporal gradients and the differential time assumption of optical flow

Superpixel labeling rate	*obstacle* ∪ *unknown* as obstacle	Only *obstacle* as obstacle
True positive rate	0.4195	0.1385
False positive rate	0.0170	0.0118

TABLE II: Quantitative results using hand-labeled ground truth. See Section IV-D for explanation of these statistics.

linearity. Iterative warping, or simply increasing the camera frame rate, are feasible remedies.

D. Quantitative Results

To quantitatively our method, we measured superpixel labeling accuracy against hand-labeled ground truth on the dataset described in Section IV-B, shown in Table II. These rates are calculated as

$$\text{True positive rate} = \frac{\text{Superpixels correctly labeled obstacle}}{\text{Superpixels that are actually obstacle}}$$

$$\text{False positive rate} = \frac{\text{Superpixels incorrectly labeled obstacle}}{\text{Superpixels that are actually clear}}$$

Intuitively, the true positive rate is the fraction of obstacle superpixels correctly labeled, and the false positive rate is the fraction of clear superpixels incorrectly labeled.

Additionally, each rate in Table II interprets the output of our method in two ways. "*obstacle* ∪ *unknown* as obstacle" counts both *obstacle* and *unknown* labels as obstacle, as is motivated by the explanation in Section IV-C, while "Only *obstacle* as obstacle" counts only *obstacle* labels as obstacle.

While superpixel labeling accuracy rates around 40% may seem low, this does not mean that only half of the obstacles are detected. Smooth texture regions on objects are often not detected while boundaries and textured regions on the same objects are. In fact, all obstacles in the dataset are detected but at varying distances, as described in Section IV-C.

Our current implementation in C++ runs at $30\,\text{frames/s}$ on 256×256 images and at $100\,\text{frames/s}$ on 128×128 images. These results were computed offline on a desktop, but the on-board computing power of the robot is similar to that of the desktop. This is several times faster than [17] due to our direct inference from spatiotemporal gradients. Our code and datasets are available online at http://borg.cc.gatech.edu/projects/autonomous-navigation-optical-flow.

V. Conclusions and Future Work

Towards the goal of autonomous obstacle avoidance for mobile robots, we have presented a method for superpixel labeling using optical flow templates. Optical flow provides a rich source of information in determining traversability, which complements image appearance and point clouds. While much past work uses optical flow for traversability via heuristics, our method instead classifies according to optical flow templates that encode typical environment shape and leverage inherent optical flow linearity. We significantly improve accuracy and efficiency over prior by labeling directly from spatiotemporal gradients and improved modeling of optical flow from obstacles. We also extend optical flow template methods to arbitrary camera optics without the need to calibrate the camera, by learning these templates from unlabeled video. Our results demonstrate successful obstacle detection in an outdoor mobile robot dataset. Future work involves increasing the obstacle detection range by further improving the optical flow model for obstacles.

References

[1] G. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. *European Conference on Computer Vision*, 2008.

[2] J. Conroy, G. Gremillion, B. Ranganathan, and J. S. Humbert. Implementation of wide-field integration of optic flow for autonomous quadrotor navigation. *Autonomous Robots*, 27(3):189–198, Aug. 2009.

[3] A. Duchon, W. H. Warren, and L. P. Kaelbling. Ecological Robotics: Controlling Behavior with Optic Flow. In *International Joint Conference on Artificial Intelligence*, 1995.

[4] A. Geiger, M. Lauer, and R. Urtasun. A generative model for 3D urban scene understanding from movable platforms. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1945–1952, June 2011.

[5] A. Giachetti, M. Campani, and V. Torre. The use of optical flow for road navigation. *IEEE Transactions on Robotics and Automation*, 14(1):34–48, 1998.

[6] J. J. Gibson. Visually controlled locomotion and visual orientation in animals. *British journal of psychology*, 49:182–194, Apr. 1958.

[7] R. Hadsell, P. Sermanet, J. Ben, A. N. Erkan, J. Han, M. K. Grimes, S. Chopra, Y. Sulsky, B. Flepp, U. Muller, and Y. LeCun. Online Learning for Offroad Robots: Using Spatial Label Propagation to Learn Long-Range Traversability. *Robotics: Science and Systems (RSS)*, 11:32, 2007.

[8] A. Huertas, L. Matthies, and A. Rankin. Stereo-Based Tree Traversability Analysis for Autonomous Off-Road Navigation. *2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION'05) - Volume 1*, pages 210–217, Jan. 2005.

[9] A. M. Hyslop and J. S. Humbert. Autonomous Navigation in Three-Dimensional Urban Environments Using Wide-Field Integration of Optic Flow. *Journal of Guidance, Control, and Dynamics*, 33(1):147–159, Jan. 2010.

[10] Y. N. Khan, P. Komma, and A. Zell. High resolution visual terrain classification for outdoor robots, 2011.

[11] D. Kim, S. M. Oh, and J. M. Rehg. Traversability classification for UGV navigation: a comparison of patch and superpixel representations. *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3166–3173, Oct. 2007.

[12] M. Lehrer, M. V. Srinivasan, S. W. Zhang, and G. A. Horridge. Motion cues provide the bee's visual world with a third dimension. *Nature*, 332(6162):356–357, Mar. 1988.

[13] D. Lieb, A. Lookingbill, and S. Thrun. Adaptive Road Following using Self-Supervised Learning and Reverse Optical Flow. *Robotics: Science and Systems (RSS)*, pages 273—-280, 2005.

[14] L. Matthies. Stereo vision for planetary rovers: Stochastic modeling to near real-time implementation. *International Journal of Computer Vision*, 8(1):71–91, July 1992.

[15] N. Nourani-Vatani, P. V. K. Borges, J. M. Roberts, and M. V. Srinivasan. Topological localization using optical flow descriptors. *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 1030–1037, Nov. 2011.

[16] A. Rieder, B. Southall, G. Salgian, R. Mandelbaum, H. Herman, P. Rander, and T. Stentz. Stereo perception on an off-road vehicle. *IEEE Intelligent Vehicle Symposium*, pages 221–226, 2002.

[17] R. Roberts and F. Dellaert. Optical Flow Templates for Superpixel Labeling in Autonomous Robot Navigation. *5th Workshop on Planning Perception and Navigation for Intelligent Vehicles (PPNIV13)*, 2013.

[18] M. Srinivasan, S. Thurrowgood, and D. Soccol. Competent Vision and Navigation Systems. *IEEE Robotics & Automation Magazine*, 16(3):59–71, Sept. 2009.

[19] P. Sturgess, K. Alahari, L. Ladicky, and P. H. S. Torr. Combining Appearance and Structure from Motion Features for Road Scene Understanding. *Procedings of the British Machine Vision Conference 2009*, pages 62.1–62.11, 2009.

[20] M. E. Tipping and C. M. Bishop. Probabilistic Principal Component Analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, Aug. 1999.

[21] A. Xu and G. Dudek. Trust-driven interactive visual navigation for autonomous robots. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3922–3929, May 2012.

[22] H. Zhang, A. Geiger, and R. Urtasun. Understanding High-Level Semantics by Modeling Traffic Patterns. *International Conference on Computer Vision (ICCV)*, 2013.