Robust Monocular Epipolar Flow Estimation

Koichiro Yamaguchi[1,2] David McAllester [1] Raquel Urtasun[1]
[1]TTI Chicago, [2]Toyota Central R&D Labs., Inc.
{yamaguchi, mcallester, rurtasun}@ttic.edu

Abstract

We consider the problem of computing optical flow in monocular video taken from a moving vehicle. In this setting, the vast majority of image flow is due to the vehicle's ego-motion. We propose to take advantage of this fact and estimate flow along the epipolar lines of the ego-motion. Towards this goal, we derive a slanted-plane MRF model which explicitly reasons about the ordering of planes and their physical validity at junctions. Furthermore, we present a bottom-up grouping algorithm which produces over-segmentations that respect flow boundaries. We demonstrate the effectiveness of our approach in the challenging KITTI flow benchmark [11] achieving half the error of the best competing general flow algorithm and one third of the error of the best epipolar flow algorithm.

1. Introduction

Optical flow is an important classical problem in computer vision, as it can be used in support of 3D reconstruction, perceptual grouping and object recognition. Here we are interested in applications to autonomous vehicles. In this setting, most of the flow can be explained by the vehicle's ego-motion. As a consequence, once the ego-motion is computed, one can treat flow as a matching problem along epipolar lines. The main difference with stereo vision resides in the fact that the epipolar lines radiate from a single epipole, called the *focus of expansion* (FOE).

A few attempts to utilize these constraints have been proposed [22], mainly in the context of scene flow (i.e., when a stereo pair is available). However, so far, we have not witnessed big performance gains by employing the epipolar constraints. In contrast, we take advantage of recent developments in stereo vision to construct robust solutions to the epipolar flow problem.

This paper has three main contributions. Our first contribution is to adapt slanted plane stereo models [39, 2] to the problem of monocular epipolar flow estimation. This allow us to exploit global energy minimization methods in order to alleviate problems in texture-less regions and produce dense flow fields. In particular, we represent the problem as one of inference in a hybrid Markov random field (MRF), where a slanted plane represents the epipolar flow for each segment and discrete random variables represent the boundary relations between each pair of neighboring segments (i.e., hinge, coplanar, occlusion). The introduction of these boundary variables allows the model to reason about ownerships of the boundary as well as to enforce physical validity of the boundary types at junctions.

In order to produce accurate results, slanted plane MRF models require a good over-segmentation of the image, where the planar assumption for each superpixel is approximately satisfied. Towards this goal, our second contribution is an efficient flow-aware segmentation algorithm in the spirit of SLIC [1], but where the segmentation energy involves both image and flow terms. This encourages the segmentation to respect both image and flow discontinuities.

The success of MRF models also depends heavily on having good data terms. Our last contribution is a local flow matching algorithm, inspired by the very successful stereo algorithm semi-global block matching [20], which computes very accurate semi-dense flow fields.

We demonstrate the effectiveness of our approach in the challenging KITTI flow benchmark [11] achieving half the error of the best competing general flow algorithm and one third of the error of the best competing epipolar flow algorithm. In the remainder of the paper, we first review related work and present our local epipolar flow algorithm. We then discuss our unsupervised segmentation algorithm which preserves epipolar flow discontinuities, and present our slanted plane MRF formulation. We conclude with our experimental evaluation and a discussion about future work.

2. Related Work

Over the past few decades we have witnessed a great improvement in performance of flow algorithms. Current approaches can be roughly divided into two categories: gradient-based approaches [21, 5, 41], which are typically based on the brightness constancy assumption, and matching-based approaches [22, 14, 25], which match a region (block) around each pixel to a set of candidate locations. *Gradient-based methods* suffer in the presence of large displacements as the brightness constancy assumption does not hold. Moreover, the regularization employed is

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.243

typically too local, yielding bad results in textureless regions. *Matching-based methods* can potentially deal with large displacements, but are typically computationally demanding due to the large amount of candidates required for good accuracy. Furthermore, they also suffer from homogeneous regions as the matching is ambiguous.

While existing many works use a variational approach for continuous flow optimization [21, 5, 6, 41], a number of recent approaches have proposed discrete MRF formulations [26, 35, 14, 25]. However, these approaches suffer from the discretization trade-off between the number of labels and the resulting computational complexity. The problem is more severe than in stereo, as instead of 1D disparities, a 2D flow field has to be discretized. [14, 25] use a coarse-to-fine approach and sampling, while [26, 35] create a set of candidate flow estimates by standard continuous optical flow algorithms.

When dealing with mostly static scenes, optical flow can be expressed as a 3D rigid motion due to the camera motion. The knowledge of this epipolar geometry has been introduced as a soft constraint in the energy function [36, 37] or as a hard constraint [33, 22]. In the latter, first the fundamental matrix is calculated and the flow estimation is formulated as a 1D search by restricting a corresponding point to lie on the epipolar line. While a soft constraint can yield less errors in independently moving objects, hard constraints can reduce computational complexity and achieve robust estimation of flow in stationary objects if the fundamental matrix is accurately estimated. In this paper we take the latter approach and adapt the highly successful slanted-plane MRF approach to stereo vision for the problem of epipolar flow estimation. As demonstrated by our experiments, this results in very significant performance gains.

3. Semi-global Block Matching for Flow

In this section we extend the popular stereo algorithm, semi-global block matching [20] to tackle the epipolar flow problem. In particular, we first convert the estimation from a 2D matching problem to a 1D search along the epipolar lines, which are defined by the vehicle's ego-motion. We then define parameterizations and cost functions which are appropriate for epipolar flow.

3.1. Epipolar Flow as a 1D Search Problem

The first step of our algorithm consists on estimating the fundamental matrix that defines the set of epipolar lines. Towards this goal, we simply match SIFT keypoints [28] in the two consecutive images, and estimate the fundamental matrix $\mathbf{F}$ using LMedS and the 8-point algorithm [15]. We then estimate the parameters of the flow that is due to camera rotation, and pose the flow problem as a 1D search along the translational flow component.

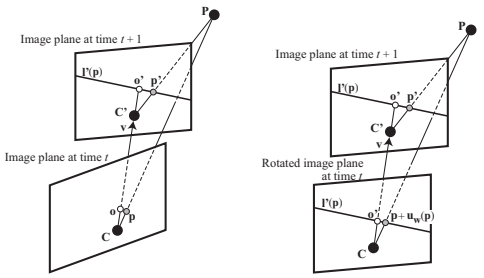

Figure 1. Epipolar flow geometry.

More formally, let $\mathbf{w} = (w_x, w_y, w_z)$ and $\mathbf{v} = (v_x, v_y, v_z)$ be the camera rotation and translation from time t to time $t+1$. Assuming that the scene is static, the flow vector $\mathbf{u} = (u_x, u_y)$ for a pixel $\mathbf{p} = (x, y)$ in the image at time t is given by

$$\mathbf{u} = \mathbf{u_w}(\mathbf{p}) + \mathbf{u_v}(\mathbf{p}, Z_\mathbf{p}), \tag{1}$$

where $\mathbf{u_w}(\mathbf{p}), \mathbf{u_v}(\mathbf{p}, Z_\mathbf{p})$ are the components of the flow due to the camera rotation and translation, respectively, and $Z_\mathbf{p}$ is the depth of pixel $\mathbf{p}$.

Assuming that the camera rotation between two images is small, $\mathbf{u_w}(\mathbf{p})$ can be expressed as follows [27],

$$\mathbf{u_w}(\mathbf{p}) = \begin{pmatrix} f w_y - w_z \bar{y} + \frac{w_y}{f}\bar{x}^2 - \frac{w_x}{f}\bar{x}\bar{y} \\ -f w_x + w_z \bar{x} + \frac{w_y}{f}\bar{x}\bar{y} - \frac{w_x}{f}\bar{y}^2 \end{pmatrix}$$

where f is the focal length of the camera and $\bar{x} = x - c_x, \bar{y} = y - c_y$, with (c_x, c_y) the principal point. Thus, we can write $\mathbf{u_w}(\mathbf{p})$ as a 5-parameter model.

$$\mathbf{u_w}(\mathbf{p}) = \mathbf{u_w}(\mathbf{p}; \mathbf{a}) = \begin{pmatrix} a_1 - a_3\bar{y} + a_4\bar{x}^2 + a_5\bar{x}\bar{y} \\ a_2 + a_3\bar{x} + a_4\bar{x}\bar{y} + a_5\bar{y}^2 \end{pmatrix}$$

An additional constraint that we can exploit to estimate the rotational component of the flow is given by the fact that $\mathbf{u_v}(\mathbf{p}, Z_\mathbf{p})$ is parallel to the epipolar line passing though that point at time $t+1$. This epipolar line is given by $\ell'(\mathbf{p}) = \mathbf{F}\tilde{\mathbf{p}}$ with $\tilde{\mathbf{p}}$ representing $\mathbf{p}$ in homogeneous coordinates. Thus, as $\mathbf{u_v}(\mathbf{p}, Z_\mathbf{p})$ being parallel to the epipolar line $\ell'(\mathbf{p})$ implies that $\mathbf{p} + \mathbf{u_w}(\mathbf{p})$ must be on $\ell'(\mathbf{p})$, we can impose that

$$\ell'(\mathbf{p})^\top (\tilde{\mathbf{p}} + \tilde{\mathbf{u}}_\mathbf{w}(\mathbf{p})) = 0 \tag{2}$$

with $\tilde{\mathbf{u}}_\mathbf{w}(\mathbf{p})$ representing $\mathbf{u_w}(\mathbf{p})$ in homogeneous coordinates. We can then estimate the parameters of the rotational flow, $\mathbf{a} = (a_1, ..., a_5)$, by minimizing the sum for all pixels of the left hand side of Eq. (2). Once this is done, we only need to estimate the flow in the direction of the epipolar lines. This is a 1D computation which is not only computationally attractive, but also results in more accurate matching, as it imposes a strong regularization.

3.2. Semi-global Block Matching for Flow

We now discuss how we can adapt the semi-global block matching stereo algorithm (SGM) [20] to estimate the translational component of flow. SGM works in three steps:

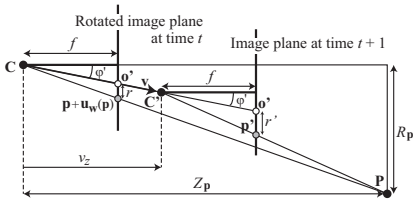

Figure 2. Geometric configuration on epipolar plane

first a pixel-wise matching cost is calculated and the cost of neighboring pixels is then aggregated taking into account smoothness constraints. Finally, postprocesing is utilized in order to obtain sub-pixel accuracy, remove spurious estimations and return only consistent estimates.

We need to define a good parameterization and a good cost function for epipolar flow. The SGM algorithm for stereo works directly on disparities. In the case of flow, using this parameterization leads to the interaction between the epipolar geometry and the scene depth, as the disparity at each point is a complex non-linear function of depth Z_p. Instead, we need to come up with a better parameterization that should be approximately linear. Towards this goal, we can write the translational component as

$$\mathbf{u_v}(\mathbf{p}, Z_\mathbf{p}) = \mathbf{e}'(\mathbf{p}) \cdot d(\mathbf{p}, Z_\mathbf{p}),$$

with $\mathbf{e}'(\mathbf{p})$ a unit vector in the direction of the epipolar line $\ell'(\mathbf{p})$ and $d(\mathbf{p}, Z_\mathbf{p})$ the disparity along the epipolar line.

Fig 1 (left) shows the epipolar geometry of two images, where $\mathbf{C}$ and $\mathbf{C}'$ are the camera centers at time t and $t+1$, $\mathbf{p}$ and $\mathbf{p}'$ are the projected image points of a 3D world point $\mathbf{P}$ and $\mathbf{o}$ and $\mathbf{o}'$ are the epipoles. Adding the rotation flow vector $\mathbf{u_w}(\mathbf{p})$ to each pixel $\mathbf{p}$ means that the image plane at time t is rotated so that its camera direction is the same as the one at time $t+1$, as shown in Fig. 1 (right). As a result, the epipole and the epipolar line in the rotated image at time t are exactly the same as those in the image at time $t+1$. Fig. 2 shows the geometric configuration on the epipolar plane, where r and r' are distances between the epipole and the projected image point on both images. We can thus write

$$r = \frac{R_\mathbf{p}}{Z_\mathbf{p}} f - f\tan\varphi' = \frac{R_\mathbf{p} - Z_\mathbf{p}\tan\varphi'}{Z_\mathbf{p}} f,$$

$$r' = \frac{R_\mathbf{p} - v_z\tan\varphi'}{Z_\mathbf{p} - v_z} f - f\tan\varphi' = \frac{R_\mathbf{p} - Z_\mathbf{p}\tan\varphi'}{Z_\mathbf{p} - v_z} f,$$

where φ' is the angle between the camera direction and the translation vector $\mathbf{v}$. We can then compute the disparity as

$$d(\mathbf{p}, Z_\mathbf{p}) = r' - r = r\frac{\frac{v_z}{Z_\mathbf{p}}}{1 - \frac{v_z}{Z_\mathbf{p}}} = |\mathbf{p} + \mathbf{u_w}(\mathbf{p}) - \mathbf{o}'|\frac{\frac{v_z}{Z_\mathbf{p}}}{1 - \frac{v_z}{Z_\mathbf{p}}}.$$

Note that the disparity is a complex function of depth.

Since the z-component v_z of the camera translation is constant for all pixels, the ratio $\frac{v_z}{Z_\mathbf{p}}$, denoted *VZ-ratio*, depends only on the distance $Z_\mathbf{p}$. As a consequence the smoothness between the VZ-ratio ($S(\omega_p, \omega_q)$ in Eq. (3)) represents the scene independent of the epipolar geometry. In order to utilize the VZ-ratio, we first quantize it as $\frac{v_z}{Z_\mathbf{p}} = \frac{\omega_\mathbf{p}}{n} v_{\max}$, with $\omega_\mathbf{p} \in \{0, 1, 2, ..., n-1\}$, where $v_{\max}$ is the maximum value of $\frac{v_z}{Z_\mathbf{p}}$ and n is the number of quantization levels. We denote $\omega_\mathbf{p}$ as the *VZ-index*.

Next, we need to define a cost function adequate for estimating the epipolar flow. We employ a cost function based on edge information as well as the Hamming distance between Census transform descriptors [40] as follows

$$C(\mathbf{p}, \omega_\mathbf{p}) = \sum_{\mathbf{q}\in\mathcal{W}(\mathbf{p})} |\mathcal{G}_t(\mathbf{q}, \mathbf{e}'(\mathbf{q})) - \mathcal{G}_{t+1}(\mathbf{q}'(\mathbf{q}, \omega_\mathbf{q}), \mathbf{e}'(\mathbf{q}))| + \lambda_{cen} \sum_{\mathbf{q}\in\mathcal{W}(\mathbf{p})} H(\mathcal{B}_t(\mathbf{q}), \mathcal{B}_{t+1}(\mathbf{q}'(\mathbf{q}, \omega_\mathbf{q})))$$

where $\mathcal{B}_t(\cdot)$ is the Census transform at time t, $H(\cdot,\cdot)$ is the Hamming distance between two binary descriptors, $\mathbf{q}'(\mathbf{q}, \omega_\mathbf{q}) = \mathbf{q} + \mathbf{u_w}(\mathbf{q}) + \mathbf{u_v}(\mathbf{q}, Z_\mathbf{q}; \omega_\mathbf{q})$ is the corresponding pixel in the second image whose VZ-index is ω_q, λ_{cen} is a constant, $\mathcal{W}(\mathbf{p})$ is a window centered at pixel $\mathbf{p}$ and $\mathcal{G}(\cdot)$ is the directional derivative in the image in the direction of the epipolar line.

The second step of SGM involves defining a cost aggregation energy. We simply define this cost as the sum of the unary cost and a smoothness term

$$E(\omega) = \sum_\mathbf{p} C(\mathbf{p}, \omega_\mathbf{p}) + \sum_{\{\mathbf{p},\mathbf{q}\}\in\mathcal{N}} S(\omega_\mathbf{p}, \omega_\mathbf{q}) \qquad (3)$$

We define $S(\omega_\mathbf{p}, \omega_\mathbf{q})$ to be 0, if $\omega_\mathbf{p} = \omega_\mathbf{q}$, and two different penalties ($0 \geq \lambda_1 \geq \lambda_2$.) depending whether they are 1 or more integers apart. Using lower penalties for small changes permits an adaptation to slanted or curved surfaces.

The flow can then be estimated by solving for the disparities $\{\omega_p\}$ by minimizing the energy in Eq. (3). While this global minimization is NP hard, in order to get a fast estimate we adopt the strategy of [20] and aggregate the matching cost in 1D from all directions equally

$$L(\mathbf{p}, \omega_p) = \sum_j L_j(\mathbf{p}, \omega_p)$$

with L_j the cost of direction j. This can be done efficiently as the minimum cost in each direction can be estimated using dynamic programming by recursively computing

$$L_j(\mathbf{p}, \omega_p) = C(\mathbf{p}, \omega_p) + \min_i \{L_j(\mathbf{p} - \mathbf{j}, i) + S(\omega_p, i)\}$$

Following [20], once Eq. (3) is minimized, we refine the VZ-index map by sub-index estimation, we remove small spurious regions, and perform a consistency check between the two consecutive frames by running the algorithm both ways and comparing the VZ values. This provides the sets $\mathcal{F}_t$ and $\mathcal{F}_{t+1}$ of the pixels, whose flow has been estimated, and VZ-indices $\hat{\omega}_t(\mathbf{p})$ and $\hat{\omega}_{t+1}(\mathbf{p}')$ of the pixels $\mathbf{p} \in \mathcal{F}_t, \mathbf{p}' \in \mathcal{F}_{t+1}$.

Algorithm 1 MotionSLIC

Init superpixels by sampling pixels in a regular grid
for $i = 1$ to #iterations **do**
  for all pixel p **do**
    $s_p = \text{argmin}_i E(\mathbf{p}, i, \theta_i, \mu_i, c_i)$
  end for
  for all superpixel s_i **do**
    $\mu_i = \frac{1}{|s_i|} \sum_{\mathbf{p} \in s_i} \mathbf{p}, \quad c_i = \frac{1}{|s_i|} \sum_{\mathbf{p} \in s_i} \mathcal{I}(\mathbf{p})$
    Compute θ_i by robust fitting a VZ-index plane
  end for
end for

Algorithm 2 PCBP Flow

Set N
$\forall i$ Initialize planes $\mathbf{y}_i^0 = (\alpha_i^0, \beta_i^0, \gamma_i^0)$ from motionSLIC
Initialize $\sigma_\alpha, \sigma_\beta$ and σ_γ
for $t = 1$ to #iters **do**
  Sample N times $\forall i$ from $\alpha_i \sim \mathcal{N}(\alpha_i^{t-1}, \sigma_\alpha)$, $\beta_i \sim \mathcal{N}(\beta_i^{t-1}, \sigma_\beta)$, $\gamma_i \sim \mathcal{N}(\gamma_i^{t-1}, \sigma_\gamma)$
  $(\mathbf{o}^t, \mathbf{y}^t) \leftarrow$ Solve discretized MRF using convex BP
  Update $\sigma_\alpha^c = \sigma_\beta^c = \frac{1}{2}\exp(-\frac{c}{10})$, $\sigma_\gamma^c = 5\exp(-\frac{c}{10})$
end for
Return $\mathbf{o}^t$, $\mathbf{y}^t$

4. Joint Segmentation and Flow Estimation

Given an estimate of the flow in a subset of the pixels, we are interested in computing an over-segmentation of the image that respects both flow and image boundaries. This over-segmentation will be used in the next section by our slanted-plane MRF model in order to produce more accurate dense flow estimations. Towards this goal, we represent the VZ-index of each superpixel with a slanted plane,

$$\omega(\mathbf{p}, \theta_{s_p}) = \alpha_{s_p} x + \beta_{s_p} y + \gamma_{s_p}, \tag{4}$$

defined with parameters $\theta_{s_p} = (\alpha_{s_p}, \beta_{s_p}, \gamma_{s_p})$, where s_p indexes the superpixel that pixel $\mathbf{p}$ belongs to. It can be shown that Eq. (4) represents a valid homography.

We frame joint unsupervised segmentation and flow estimation as an energy minimization problem, and define the energy of each pixel as the sum of energies encoding shape, appearance and flow, taking special care into modeling occlusions. The input to our algorithm is the two images as well as our initial (possibly sparse) flow estimate $\hat{\omega}$ (see section 3). We now discuss each energy term in more details.

Regular Shape: We prefer superpixels that have a regular shape. Following [1] we encode this as

$$E_{\text{pos}}(\mathbf{p}, \mu_{s_p}) = \left|\left|\mathbf{p} - \mu_{s_p}\right|\right|_2^2 / g$$

where μ_{s_p} is the superpixel centroid, $g = W \times H/m$ with W, H the width and height of the image and m the desired number of superpixels.

Appearance: We encourage the elements of the same superpixel to have similar appearance. We do so by defining

$$E_{\text{col}}^t(\mathbf{p}, c_{s_p}) = \left(\mathcal{I}_t(\mathbf{p}) - c_{s_p}\right)^2$$

where c_{s_p} is the mean appearance descriptor for superpixel s_p. Let $\mathbf{q}(\mathbf{p}, \theta_{s_p})$ be the predicted location of pixel $\mathbf{p}$ at time $t+1$ computed using the plane assumption (Eq. (4)) and (1). We can encode a similar energy for the next frame

$$E_{\text{col}}^{t+1}(\mathbf{p}, c_{s_p}, \theta_{s_p}) = \begin{cases} \left(\mathcal{I}_{t+1}(\mathbf{q}(\mathbf{p}, \theta_{s_p})) - c_{s_p}\right)^2 & \text{if } \delta_{\text{no}}(\mathbf{p}, \theta_{s_p}) \\ E_{\text{col}}^t(p, c_{s_p}) & \text{otherwise} \end{cases}$$

where $\delta_{\text{no}}(\mathbf{p}, \theta_{s_p})$ represents whether a pixel is non-occluded. We say that a pixel $\mathbf{p}$ is non-occluded if we have an initial estimate, i.e., $\mathbf{p} \in \mathcal{F}_t$ or if $\mathbf{q}(\mathbf{p}, \theta_{s_p}) \in \mathcal{F}_{t+1}$ and $\omega(\mathbf{p}, \theta_{s_p}) \geq \hat{\omega}_{t+1}(\mathbf{q}(\mathbf{p}, \theta_{s_p}))$.

Flow: This potential enforces that the plane parameters should agree with the input flow $\hat{\omega}_t(\mathbf{p})$ as follows

$$E_{\text{disp}}^t(\mathbf{p}, \theta_{s_p}) = \begin{cases} \left(\omega(\mathbf{p}, \theta_{s_p}) - \hat{\omega}_t(\mathbf{p})\right)^2 & \text{if } \mathbf{p} \in \mathcal{F}_t \\ \lambda_{\text{d}} & \text{otherwise} \end{cases}$$

with λ_{d} a constant. Additionally, we define

$$E_{\text{disp}}^{t+1}(\mathbf{p}, \theta_{s_p}) = \begin{cases} \left(\omega(\mathbf{p}, \theta_{s_p}) - \hat{\omega}_{t+1}(\mathbf{q}(\mathbf{p}, \theta_{s_p}))\right)^2 & \text{if } \delta_{\text{no}}(\mathbf{p}, \theta_{s_p}) \\ \lambda_{\text{d}} & \text{otherwise} \end{cases}$$

We can define the total energy of a pixel as

$$E = E_{\text{col}}^t(\mathbf{p}, c_{s_p}) + E_{\text{col}}^{t+1}(\mathbf{p}, c_{s_p}, \theta_{s_p}) + \lambda_{\text{pos}} E_{\text{pos}}(\mathbf{p}, \mu_{s_p}) + \lambda_{\text{disp}} \left\{ E_{\text{disp}}^t(\mathbf{p}, \theta_{s_p}) + E_{\text{disp}}^{t+1}(\mathbf{p}, \theta_{s_p}) \right\},$$

where λ_{pos} and λ_{disp} are two scalars. The problem of joint unsupervised segmentation and flow estimation becomes

$$\min_{\Theta, \mathbf{S}, \mu, \mathbf{c}} \sum_{\mathbf{p}} E(\mathbf{p}, s_p, \theta_{s_p}, \mu_{s_p}, c_{s_p}). \tag{5}$$

where $\mathbf{S} = \{s_1, \cdots, s_m)$ is the set of superpixel assignments, $\Theta = \{\theta_1, \cdots, \theta_m\}$ the set of plane parameters, $\mu = \{\mu_1, \cdots, \mu_m\}$ the mean location of each superpixel, and $\mathbf{c} = \{c_1, \cdots, c_m\}$ the mean appearance descriptor.

This is a non-convex mixed continuous-discrete optimization problem, which is NP-hard to solve. We derive an iterative scheme that works in three steps: first we minimize the energy with respect to the assignments, we update the parameters μ_{s_p}, c_{s_p} by simply computing their means, and then compute the plane parameters by using a robust estimator. The algorithm, which we denote *MotionSLIC*, is summarized in Algorithm 1. This algorithm can be extended to stereo vision by simply replacing the two consecutive frames with the left and right images of the stereo pair, and the VZ-ratio with disparity. Our experimental evaluation will demonstrate the effectiveness of our method in both epipolar flow and stereo estimation problems.

	> 2 pixels		> 3 pixels		> 4 pixels		> 5 pixels		End-Point	
	Non-Occ	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**
Pyramid-LK [3]	72.46 %	75.91 %	65.74 %	70.09 %	60.99 %	65.98 %	57.22 %	62.72 %	21.7 px	33.1 px
OCV-BM [4]	65.60 %	70.03 %	63.46 %	68.16 %	61.85 %	66.75 %	60.41 %	65.49 %	24.4 px	33.3 px
PolyExpand [10]	50.16 %	56.39 %	47.54 %	53.95 %	45.88 %	52.34 %	44.53 %	51.03 %	17.2 px	25.2 px
HAOF [5]	38.19 %	45.68 %	35.76 %	45.36 %	33.98 %	41.61 %	32.48 %	40.12 %	11.1 px	18.2 px
GCSF [7]	39.53 %	47.25 %	33.23 %	41.74 %	29.24 %	38.23 %	26.33 %	35.64 %	7.0 px	15.3 px
DB-TV-L1 [41]	33.87 %	42.00 %	30.75 %	39.13 %	28.42 %	36.94 %	26.50 %	35.10 %	7.8 px	14.6 px
C+NL [34]	26.42 %	35.28 %	24.64 %	33.35 %	23.53 %	32.06 %	22.71 %	31.08 %	9.0 px	16.4 px
LDOF [6]	24.43 %	33.87 %	21.86 %	31.31 %	20.13 %	29.48 %	18.72 %	27.97 %	5.5 px	12.4 px
RSRS-Flow [13]	22.68 %	31.81 %	20.74 %	29.68 %	19.55 %	28.24 %	18.65 %	27.13 %	6.2 px	12.1 px
HS [21]	22.02 %	31.18 %	19.92 %	28.86 %	18.60 %	27.28 %	17.61 %	26.07 %	5.8 px	11.7 px
GC-BM-Mono [22]	24.79 %	34.59 %	19.49 %	29.88 %	17.04 %	27.56 %	15.42 %	25.93 %	5.0 px	12.1 px
GC-BM-Bino [22]	23.07 %	33.10 %	18.93 %	29.37 %	16.80 %	27.32 %	15.31 %	25.80 %	5.0 px	12.0 px
TGV2CENSUS [38]	13.33 %	21.11 %	11.14 %	18.42 %	9.98 %	16.83 %	9.19 %	15.68 %	2.9 px	6.6 px
fSGM [18]	14.56 %	25.90 %	11.03 %	22.90 %	9.43 %	21.50 %	8.44 %	20.63 %	3.2 px	12.2 px
Ours	**6.33 %**	**11.59 %**	**4.08 %**	**8.70 %**	**3.14 %**	**7.23 %**	**2.58 %**	**6.26 %**	**0.9 px**	**2.2 px**

Table 1. **Flow**: Comparison with the state of the art on the test set of KITTI [11].

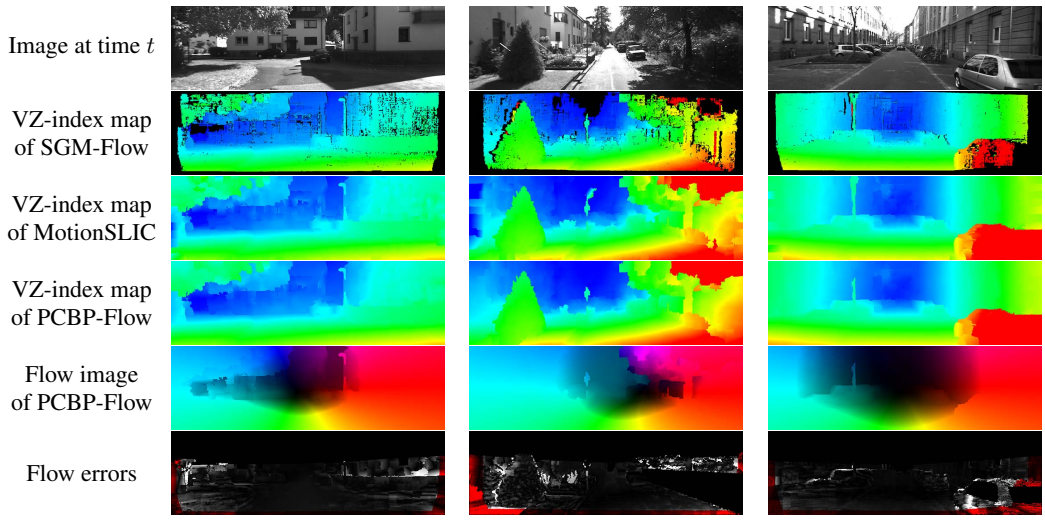

Figure 3. KITTI examples

5. Slanted-plane MRFs for Epipolar Flow

Slanted-plane MRF models are among the leading approaches to stereo vision [2]. Recently, [39] proposed a slanted-plane MRF model for stereo vision that reasons about segments as well as occlusion boundaries. Here we follow a similar idea, and represent the epipolar flow estimation problem as inference in a mixed continuous-discrete random field. The continuous variables represent 3D planes encoding the VZ-ratio, while the discrete variables encode the type of boundaries between pairs of superpixels. Our approach takes as input epipolar flow as well as an over-segmentation of the image. In particular, we employ the epipolar flow fields and segmentations estimated by MotionSLIC (see section 4).

Let $y_i = (\alpha_i, \beta_i, \gamma_i) \in \Re^3$ be a random variable representing the $i-$th slanted plane. For each pixel $\mathbf{p}$ belonging to the $i-$th segment, we can compute its VZ-ratio as

$$\bar{\omega_i}(\mathbf{p}, \mathbf{y}_i) = \alpha_i(u - c_{iu}) + \beta_i(v - c_{iv}) + \gamma_i \qquad (6)$$

where $\mathbf{p} = (u, v)$, $\mathbf{c}_i = (c_{iu}, c_{iu})$ is the center of the i-th segment, γ_i the VZ-ratio in the segment center, and $\mathbf{y}_i$ represents the slanted plane $y_i = (\alpha_i, \beta_i, \gamma_i)$. We have centered the planes as it improves the efficiency. Let $o_{i,j} \in \{co, hi, lo, ro\}$ be a discrete random variable representing whether two neighboring planes are coplanar, form a hinge or an occlusion boundary. Here, lo implies that plane i occludes plane j, and ro the opposite. We define our hybrid conditional random field in terms of all slanted-planes and boundary variables and encode potentials over sets of continuous, discrete or mixture of both types of variables. We now briefly describe the potentials employed, and refer the reader to [39] for more details.

VZ-ratio: We define truncated quadratic potentials for each segment encoding that the plane should agree with the epipolar flow estimated using the algorithm from section 3.

Boundary: We employ 3-way potentials linking our discrete and continuous variables expressing the fact that when

	> 2 pixels		> 3 pixels		> 4 pixels		> 5 pixels		End-Point	
	Non-Occ	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**
Ours SGM-Flow	7.32 %	17.74 %	4.72 %	14.55 %	3.58 %	12.73 %	2.93 %	11.43 %	1.0 px	3.7 px
Ours MotionSLIC	6.72 %	14.06 %	4.36 %	10.91 %	3.34 %	9.24 %	2.74 %	8.12 %	1.0 px	2.7 px
Ours PCBP-Flow	6.33 %	11.59 %	4.08 %	8.70 %	3.14 %	7.23 %	2.58 %	6.26 %	0.9 px	2.2 px

Table 2. **Importance of each step** on the test set of KITTI [11].

	> 2 pixels		> 3 pixels		> 4 pixels		> 5 pixels		End-Point	
	Non-Occ	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**
disparity	8.41 %	19.07 %	5.66 %	16.21 %	4.26 %	14.56 %	3.33 %	13.33 %	1.2 px	5.0 px
VZ-index	**7.11 %**	**16.75 %**	**4.76 %**	**13.71 %**	**3.58 %**	**11.81 %**	**2.78 %**	**10.35 %**	**1.0 px**	**3.1 px**

Table 3. **Use of VZ index vs disparity** on SGM-Flow evaluated on the validation set of KITTI [11].

two neighboring planes are hinge or coplanar they should agree on the boundary, and when a segment occludes another, the boundary should be explained by the occluder.

Compatibility: We penalize occlusion boundaries that are not supported by the data. Additionally, we define a potential that penalizes negative VZ-ratios.

Occam's razor: We impose a regularization on the type of occlusion boundary, where we prefer simpler explanations (i.e., coplanar better than hinge better than occlusion).

Junction Feasibility: We encode the physical validity of junctions of 3 and 4 planes. Although these potentials are high-order, they only involve variables with 4 states, thus the additional complexity is not prohibitive.

Color similarity: This potential encodes the fact that we expect segments which are coplanar to have similar color statistics, while the entropy is higher when the planes form an occlusion boundary or a hinge. We employ the χ-squared distance between histograms of neighboring segments.

Computing the MAP estimate of our hybrid MRF is NP-hard. Instead, we rely on approximate algorithms based on LP relaxations. Following [39] we make use of particle convex belief propagation (PCBP) [29], a technique that is guaranteed to converge and gradually approach the optimum. PCBP is an iterative algorithm that works as follows: For each continuous variable particles are sampled around the current solution. These samples act as labels in a discretized MRF which is solved to convergence using convex belief propagation [16]. The current solution is then updated with the MAP estimate obtained on the discretized MRF. This process is repeated for a fixed number of iterations. In our implementation, we use the distributed message passing algorithm of [32] to solve the discretized MRF at each iteration. Algorithm 2 depicts our *PCBP-Flow* algorithm. At each iteration, to balance the trade off between exploration and exploitation, we decrease the variance of the distribution we sample from. Following [39], we discretize the continuous variables, and utilize the algorithm of [17] for learning the importance of each potential.

6. Experimental Evaluation

We perform our experiments on the challenging KITTI dataset [11], which is composed of 194 training and 195 test high-resolution images captured from an autonomous driving platform driving around a urban environment. We use 10 images for training and 184 for validation. The ground truth is semi-dense covering approximately 30 % of the pixels. We employ two different metrics to evaluate our approach. The first one measures the average number of pixels (non-occluded and all) whose error is bigger than a fixed threshold. The second one reports end-point error for both settings. For all experiments, we employ the same parameters which have been validated on the validation set. We use $\upsilon_{\max} = 0.3$ and $n = 256$ for our discretization of the VZ-ratio. For SGM-Flow, we set $\lambda_{cen} = 0.5$, $\lambda_1 = 100$, $\lambda_2 = 1600$, use a window $\mathcal{W}(\mathbf{p})$ of size 5×5, and aggregate information over 4 paths. Unless otherwise stated, for MotionSLIC we set the number of superpixels $m = 400$, $\lambda_{\text{pos}} = 4000, \lambda_{\text{disp}} = 30, \lambda_{\text{d}} = 3$, use 10 iterations and a Lab vector as the mean color representation. For PCBP, we employ the same parameter values as [39], and run inference with 10 particles and 5 iterations of re-sampling.

Comparison with the state-of-the-art: We compare our approach to the state-of-the-art in the test set of KITTI. As shown in Table 1, our approach significantly outperforms all approaches, yielding approximately half the error of the best general flow algorithm, and a third of the error of the best epipolar flow algorithm, i.e., GC-BM-Mono [22]. Interestingly, even a scene flow approach, i.e., GC-BM-Bino [22] that unlike our approach utilizes stereo pairs results in three times more error. Fig. 3 depicts our flow estimations.

Importance of each step: We evaluate the importance of each step of our pipeline. Table 2 depicts errors of our SGM-Flow (section 3), our MotionSLIC (section 4) as well as our PCBP-Flow (section 5) algorithms. Note that the output of SGM-flow is used as input for motionSLIC, and the output of motionSLIC is used as input to PCBP-flow. Each step significantly improves results.

Running Time: We evaluate the run time of our approach. KITTI images have on average 1237×374 pixels. SGM-Flow takes on average 5.7s per image, 1.5s for MotionSLIC and 3.5 minutes for PCBP-Flow. Thus state-of-the-art estimates can be obtained in only a few seconds, as SGM-Flow and MotionSLIC significantly outperforms

	> 2 pixels		> 3 pixels		> 4 pixels		> 5 pixels		End-Point	
	Non-Occ	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**
Oracle FOE	0.000 %	0.000 %	0.000 %	0.000 %	0.000 %	0.000 %	0.000 %	0.000 %	0.02 px	0.03 px
Estimated FOE	0.33 %	0.48 %	0.17 %	0.22 %	0.06 %	0.10 %	0.00 %	0.02 %	0.2 px	0.2 px
Oracle GT	1.46 %	1.77 %	1.14 %	1.33 %	0.98 %	1.12 %	0.87 %	0.98 %	0.3 px	0.4 px
Oracle estimated	2.47 %	3.15 %	1.75 %	2.12 %	1.39 %	1.64 %	1.18 %	1.37 %	0.5 px	0.6 px
Our SGM-Flow	6.11 %	10.97 %	4.08 %	8.22 %	3.08 %	6.67 %	2.38 %	5.58 %	0.9 px	1.8 px

Table 4. **Epipolar constraint and piece-wise planar assumptions** on the validation set of KITTI [11]. If the true FOE is used to estimate flow, there is no error ("Oracle FOE"). The error of the best oracle match along the epipolar line when employing our estimated FOE is also very small. In "Oracle GT", ground truth flow vectors are converted into VZ-index values using the epipolar lines estimated from ground truth, and VZ-index planes are fitted to the superpixel segments, which are generated by motionSLIC. In "Oracle estimated", flow vectors of ground truth are converted to VZ-index values using our estimated epipolar lines. SGM-Flow has no oracle access.

	> 2 pixels		> 3 pixels		> 4 pixels		> 5 pixels		End-Point	
	Non-Occ	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**
pos + 1-color	8.02 %	14.37 %	5.68 %	11.22 %	4.46 %	9.35 %	3.59 %	8.01 %	1.1 px	2.4 px
pos + 1-color + 1-flow	6.63 %	13.26 %	4.50 %	10.33 %	3.44 %	8.61 %	2.69 %	7.36 %	0.9 px	2.3 px
pos + 1-color + 2-flow	6.53 %	13.17 %	4.42 %	10.26 %	3.37 %	8.55 %	2.62 %	7.31 %	0.9 px	2.3 px
pos + 2-color + 2-flow	6.53 %	13.19 %	4.39 %	10.25 %	3.34 %	8.53 %	2.59 %	7.28 %	0.9 px	2.3 px

Table 5. **Importance of energy terms of MotionSLIC:** The first number in color and flow denotes the number of images used.

existing approaches.

Importance of VZ-index: As shown in Table 3 using VZ-index instead of disparity as parameterization in our SGM-Flow algorithm significantly improves performance.

Oracle: We would like to estimate how much we loose due to the assumptions of our model. Our first assumption is that most of the flow is due to the ego-motion. As shown in Table 4, if the true FOE is used to estimate flow, there is basically no error ("Oracle FOE"). When utilizing our estimated FOE (via SIFT matching and 8-point algorithm with RANSAC), the error of the best oracle match along the epipolar line is also very small. Thus, even with a noisy egomotion estimation, one could potentially achieve very low error. The second assumption is that the VZ-ratio is piece-wise planar. Note that given the ground truth epipolar lines ("Oracle GT"), the piece-wise planar assumption is fairly accurate. When the epipolar lines are estimated by our ego-motion estimation ("Oracle estimated"), the piece-wise planar assumption becomes worse, but is still a good fit. Our algorithm ("SGM-Flow") is not far from the oracle.

Energy terms in MotionSLIC: Table 5 depicts performance as a function of the energy terms employed. The first row coincides with SLIC [1]. Note that performance significantly increases by adding flow.

Stereo: Our MotionSLIC algorithm can be utilized for stereo vision in order to compute disparities and segmentations that respect depth boundaries. We called this StereoSLIC. Once computed, it can be used as input to the slanted-plane MRF model of [39]. We call this PCBP-StereoSLIC. As shown in Table 6 both algorithms outperform the state-of-the-art. Importantly, StereoSLIC requires only a few seconds per image.

7. Conclusion and Future Work

We have presented a slanted-plane MRF model for the problem of epipolar flow estimation which utilizes a robust data term as well as an over-segmentation of the image that respects flow boundaries. We have demonstrated the effectiveness of our approach in the challenging KITTI flow benchmark, achieving half the error of the best competing general flow algorithm and one third of the error of the best competing epipolar flow algorithm. Our algorithms can be easily parallelized (see e.g., [32]). We plan to explore this to achieve real-time performance in the future.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC superpixels. *Tech rep.*, 2010. 1, 4, 7

[2] M. Bleyer, C. Rother, and P. Kohli. Surface stereo with soft segmentation. In *CVPR*, 2010. 1, 5

[3] J.-Y. Bouguet. Pyramidal implementation of the lucas kanade feature tracker. *Intel*, 2000. 5

[4] G. Bradski. The OpenCV library. *Dr. Dobb's Journal of Software Tools*, 2000. 5, 8

[5] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *ECCV*, 2004. 1, 2, 5

[6] T. Brox and J. Malik. Large displacement optical flow: Descriptor matching in variational motion estimation. *PAMI*, 2011. 2, 5

[7] J. Cech, J. Sanchez-Riera, and R. P. Horaud. Scene flow estimation by growing correspondence seeds. In *CVPR*, 2011. 5, 8

[8] J. Cech and R. Sara. Efficient sampling of disparity space for fast and accurate matching. In *BenCOS*, 2007. 8

[9] N. Einecke and J. Eggert. A two-stage correlation method for stereoscopic depth estimation. In *DICTA*, 2010. 8

[10] G. Farneback. Two-frame motion estimation based on polynomial expansion. In *SCIA*, 2003. 5

	> 2 pixels		> 3 pixels		> 4 pixels		> 5 pixels		End-Point	
	Non-Occ	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**	**Non-Occ**	**All**
GC+occ [23]	39.76 %	40.97 %	33.50 %	34.74 %	29.86 %	31.10 %	27.39 %	28.61 %	8.6 px	9.2 px
OCV-BM [4]	27.59 %	28.97 %	25.39 %	26.72 %	24.06 %	25.32 %	22.94 %	24.14 %	7.6 px	7.9 px
CostFilter [31]	25.85 %	27.05 %	19.96 %	21.05 %	17.12 %	18.10 %	15.51 %	16.40 %	5.0 px	5.4 px
GCS [8]	18.99 %	20.30 %	13.37 %	14.54 %	10.40 %	11.44 %	8.63 %	9.55 %	2.1 px	2.3 px
GCSF [7]	20.75 %	22.69 %	13.02 %	14.77 %	9.48 %	11.02 %	7.48 %	8.84 %	1.9 px	2.1 px
SDM [24]	15.29 %	16.65 %	10.98 %	12.19 %	8.81 %	9.87 %	7.44 %	8.39 %	2.0 px	2.3 px
ELAS [12]	10.95 %	12.82 %	8.24 %	9.95 %	6.72 %	8.22 %	5.64 %	6.94 %	1.4 px	1.6 px
OCV-SGBM [20]	10.58 %	12.20 %	7.64 %	9.13 %	6.04 %	7.40 %	5.04 %	6.25 %	1.8 px	2.0 px
ITGV [30]	8.86 %	10.20 %	6.31 %	7.40 %	5.06 %	5.97 %	4.26 %	5.01 %	1.3 px	1.5 px
SNCC [9]	10.62 %	11.86 %	6.27 %	7.33 %	4.68 %	5.58 %	3.85 %	4.62 %	1.4 px	1.5 px
SGM [20]	8.81 %	10.31 %	5.83 %	7.08 %	4.43 %	5.45 %	3.58 %	4.43 %	1.2 px	1.3 px
iSGM [19]	8.04 %	10.09 %	5.16 %	7.19 %	3.87 %	5.84 %	3.15 %	5.03 %	1.2 px	2.1 px
SGBM [39]	6.88 %	8.19 %	4.49 %	5.54 %	3.35 %	4.20 %	2.66 %	3.35 %	0.9 px	1.1 px
PCBP [39]	6.25 %	7.78 %	4.13 %	5.45 %	3.18 %	4.32 %	2.66 %	3.66 %	0.9 px	1.2 px
Our StereoSLIC	**5.90 %**	**7.33 %**	**3.99 %**	**5.17 %**	**3.08 %**	**4.07 %**	**2.51 %**	**3.35 %**	**0.9 px**	**1.0 px**
Our PCBP-StereoSLIC	**5.36 %**	**6.90 %**	**3.49 %**	**4.79 %**	**2.66 %**	**3.78 %**	**2.20 %**	**3.16 %**	**0.8 px**	**1.0 px**

Table 6. **Stereo**: Comparison with the state-of-the-art on the test set of KITTI [11].

[11] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? In *CVPR*, 2012. 1, 5, 6, 7, 8

[12] A. Geiger, M. Roser, and R. Urtasun. Efficient large-scale stereo matching. In *ACCV*, 2010. 8

[13] P. Ghosh and B. Manjunath. Robust simultaneous registration and segmentation with sparse error reconstruction. *PAMI*, 2012. 5

[14] B. Glocker, N. Paragios, N. Komodakis, G. Tziritas, and N. Navab. Optical flow estimation with uncertainties through dynamic MRFs. In *CVPR*, 2008. 1, 2

[15] R. Hartley. In defense of the eight-point algorithm. *PAMI*, 1997. 2

[16] T. Hazan and A. Shashua. Norm-product belief propagation: Primal-dual message-passing for approximate inference. *IEEE Trans. Information Theory*, 2010. 6

[17] T. Hazan and R. Urtasun. A primal-dual message-passing algorithm for approximated large scale structured prediction. *NIPS*, 2010. 6

[18] S. Hermann and R. Klette. Hierarchical scan line dynamic programming for optical flow using semi-global matching. In *Intelligent Mobile Vision, ACCV-Workshop*, 2012. 5

[19] S. Hermann and R. Klette. Iterative semi-global matching for robust driver assistance systems. In *ACCV*, 2012. 8

[20] H. Hirschmueller. Stereo processing by semiglobal matching and mutual information. *PAMI*, 2008. 1, 2, 3, 8

[21] B. K. P. Horn and B. G. Schunck. Determining optical flow: A retrospective. *AI*, 1993. 1, 2, 5

[22] B. Kitt and H. Lategahn. Trinocular optical flow estimation for intelligent vehicle applications. In *ITSC*, 2012. 1, 2, 5, 6

[23] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions using graph cuts. In *ICCV*, 2001. 8

[24] J. Kostkova. Stratified dense matching for stereopsis in complex scenes. In *BMVC*, 2003. 8

[25] C. Lei and Y. Yang. Optical flow estimation on coarse-to-fine region-trees using discrete optimization. In *ICCV*, 2009. 1, 2

[26] V. Lempitsky, S. Roth, and C. Rother. FusionFlow: Discrete-continuous optimization for optical flow estimation. In *CVPR*, 2008. 2

[27] H. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. In *Proc. R. Soc. Lond. B*, 1980. 2

[28] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 2

[29] J. Peng, T. Hazan, D. McAllester, and R. Urtasun. Convex max-product algorithms for continuous MRFs with applications to protein folding. In *ICML*, 2011. 6

[30] R. Ranftl, S. Gehrig, T. Pock, and H. Bischof. Pushing the Limits of Stereo Using Variational Stereo Estimation. In *IEEE Intelligent Vehicles Symposium*, 2012. 8

[31] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. In *CVPR*, 2011. 8

[32] A. Schwing, T. Hazan, M. Pollefeys, and R. Urtasun. Distributed message passing for large scale graphical models. In *CVPR*, 2011. 6, 7

[33] N. Slesareva, A. Bruhn, and J. Weickert. Optic flow goes stereo: a variational method for estimating discontinuity-preserving dense disparity maps. In *DAGM*, 2005. 2

[34] D. Sun, S. Roth, and M. J. Black. Secrets of optical flow estimation and their principles. In *CVPR*, 2010. 5

[35] W. Trobin, T. Pock, D. Cremers, and H. Bischof. Continuous energy minimization via repeated binary fusion. In *ECCV*, 2008. 2

[36] L. Valgaerts, A. Bruhn, and J. Weickert. A variational model for the joint recovery of the fundamental matrix and the optical flow. In *DAGM*, 2008. 2

[37] A. Wedel, D. Cremers, T. Pock, and H. Bischof. Structure- and motion-adaptive regularization for high accuracy optic flow. In *ICCV*, 2009. 2

[38] M. Werlberger. *Convex Approaches for High Performance Video Processing*. Phdthesis, 2012. 5

[39] K. Yamaguchi, T. Hazan, D. McAllester, and R. Urtasun. Continuous markov random fields for robust stereo estimation. In *ECCV*, 2012. 1, 5, 6, 7, 8

[40] R. Zabih and J. Woodfill. Non-parametric local transforms for computing visual correspondence. In *ECCV*, 1994. 3

[41] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime TV-L1 optical flow. In *DAGM*, 2007. 1, 2, 5