Occlusion Detection and Motion Estimation via Convex Optimization*

Michalis Raptis†, Alper Ayvaci† and Stefano Soatto

June 22, 2010
Revised: February 22, 2011

Abstract

We tackle the problem of detecting occluded regions in a video stream. Under assumptions of Lambertian reflection and static illumination, the task can be posed as a variational optimization problem, and its solution approximated using convex minimization. We describe efficient numerical schemes that reach the global optimum of the relaxed cost functional, for any number of independently moving objects, and any number of occlusion layers. We test the proposed algorithm on benchmark datasets, expanded to enable evaluation of occlusion detection performance, in addition to optical flow.

1 Introduction

Occlusion phenomena are a critical component of the image formation process, and play a role in shaping the statistics of natural images, in priming visual recognition of detached objects, in navigation and interaction with natural objects and environments. Occlusions arise when a portion of the scene is visible in one image, but not another. In Da Vinci Steropsis, portions of the scene that are visible from the left eye are not visible from the right eye, and vice-versa. In a video-stream, occlusions occur at depth discontinuities. We are interested in determining the *occluded regions*, that is the subset of an image domain that back-projects onto portions of the scene that are not *co-visible* from a temporally adjacent image.[1] The occluded region is, in general, multiply connected, and can be quite complex, as the example of a barren tree illustrates.

Portions of the scene that are *co-visible* can be mapped onto one another by a domain deformation [34], called *optical flow*. It is, in general, different from the *motion field*, that is the projection onto the image plane of the spatial velocity of the scene [37], unless three conditions are satisfied: (a) Lambertian reflection, (b) constant illumination, and (c) constant visibility properties of the scene. Most surfaces with benign reflectance properties (diffuse/specular) can be approximated as Lambertian almost everywhere under sparse illuminants (e.g., the sun). In any case, widespread violation of Lambertian reflection does not enable correspondence, so most optical flow methods embrace (a), either implicitly or explicitly. Similarly, constant illumination (b) is a reasonable assumption for ego-motion (the scene is not moving relative to the light source), and even for objects moving (slowly) relative to the light source. Assumption (c) is needed in order to have a dense flow field: If an image contains portions of its domain that are *not visible* in another image, these can patently *not* be mapped onto it by optical flow vectors; (c) is often assumed because optical flow is defined in the limit where two images are sampled infinitesimally close in time, in which case *there are no occluded regions,* and one can focus solely on discontinuities[2] of the motion field. Thus, the great majority of

*Research supported by AFOSR, ARO, and ONR.

†M. Raptis and A. Ayvaci contributed equally to this work.

[1] This process could be generalized to global co-visibility, resulting in a model of the world with topologically distinct “layers” [38]. This is beyond the scope of this paper and has already been addressed in a variational setting, the first example being [18, 17].

[2] In occluded regions, the problem is *not* that optical flow is discontinuous; it is simply not defined; *it does not exist.* Motion in occluded regions can be *hallucinated* or *extrapolated*, based on the prior or regularizer. However, whatever motion is assigned to an occluded region *cannot be validated from the data.*

variational motion estimation approaches provide an estimate of a dense flow field, defined at each location on the image domain, *including occluded regions.* In their defense, it can be argued that for small parallax (slow-enough motion, or far-enough objects, or fast-enough temporal sampling) occluded areas are small. However, small does not mean absent, nor unimportant, as occlusions are critical to perception [13] and a key for developing representations for recognition. For this reason, *we focus on occlusion detection* in video streams.

Occlusion detection would be easy if the motion field was known. Vice-versa, optical flow estimation would be easy if the occluded domain was known. As often in vision problems, one knows neither, so in the process of inferring the object of inference (the occluded domain) we will estimate optical flow, the "nuisance variable," as a byproduct.

In this manuscript we (I) show that, starting from the standard assumptions (a)-(b), the problem of detecting (multiply-connected) occlusion regions can be formulated as a variational optimization problem (section 2). We then (II) show how the functional to be minimized can be relaxed into a sequence of convex functionals and minimized using *re-weighted* ℓ_1 optimization (appendix A and eq. (16)). At each iteration, the functional to be minimized is related to those used for optical flow estimation, but the minimization is with respect to the indicator function of the occluded region, not just the (dense) optical flow field. We then bring to bear two different approaches to optimize these functionals, one is (III) an optimal first-order method, due to Nesterov (section 3), and the other one is (IV) an alternating minimization technique, known as split-Bregman method (section 4). We evaluate our approach empirically in sections 5 and 1.2, and discuss its strengths and limitations of in section 6.

To the best of our knowledge, neither the formulation of occlusion detection and motion estimation as a joint minimization problem under sparsity prior on the occluded regions (I), nor the use of re-weighted ℓ_1 (II), Nesterov's algorithm (III), or split-Bregman method (IV) have ever been presented before in the optical flow literature. A preliminary conference version of this paper has appeared in [2]. The current version provides an additional optimization method, split-Bregman, for improving the computation speed. We have also included extensive experiments analyzing re-weighting steps and compares proposed method to robust flow estimation methods.

1.1 Prior Related Work

Several algorithms have been proposed recently to tackle the issue of occlusion detection. One class of methods define occlusion as the region of mismatch in forward and backward motion estimation. Proesmans et al. [25] and Alvarez et al. [1] explicitly detected occlusions with this approach, however, they did not put any effort to fix the motion estimates affected by outliers (occlusions). Others [21, 20, 33] formulated joint motion estimation and occlusion detection problem as a NP-hard problem in a discrete setting, and find an approximate solution with a combinatorial optimization algorithm. Others [16, 5] also exploited motion symmetry to detect occlusions and suppress the effects of occlusions on the data term by weighting it with a monotonically decreasing function. These methods rely on the prior to "inpaint" the optical flow in the occluded region (where correspondence cannot be established) and then test for inconsistencies in the inpainted flow.

A second class of methods uses the residual from optical flow estimation to decide whether a region is occluded. Strecha et al. [31] proposed a probabilistic formulation to detect occluded regions using the estimated noise model and histogram of occluded pixel intensities. Xiao et al. [44] threshold the residual obtained using level-set methods to find occluded areas. Both try to minimize a non-convex energy function by iterating between two subproblems, occlusion detection and motion estimation.

Occlusion phenomena have also been a concern in the optical flow community since the first global formulation was proposed by Horn and Schunck [15]. Black and Anandan [6] proposed replacing the ℓ_2 norm of the residual with a non-convex Lorentzian penalty. Another common criterion used for this purpose is the ℓ_1 norm of the residual, which is non-trivial to minimize since it is non-smooth. [8, 7] use Charbonnier's penalty that is a differentiable approximation of the ℓ_1 norm; others [41, 40, 43] solved the non-smooth problem with primal-dual methods decoupling the matching and regularization terms. However, none of these robust flow estimation methods focus on the detection of occlusions.

1.2 Evaluation

Optical flow estimation is a mature area of computer vision, and benchmark datasets have been developed, the best known example being the Middlebury [3]. Unfortunately, no existing benchmark provides ground truth for occluded regions, nor a scoring mechanism to evaluate the performance of occlusion detection algorithms. Unfortunately, this also biases the motion estimation scoring mechanism as ground truth motion is provided on the entire image domain, *including occluded regions,* where it can be *extrapolated* using the priors/regularizers, but not validated from the data.

To overcome this gap, we have produced a new benchmark by taking a subset of the training data in the Middlebury dataset, and hand-labeling occluded regions. We then use the same evaluation method of the Middlebury *for the (ground truth) regions that are co-visible* in at least two images. This provides a motion estimation score. Then, we provide a separate score for occlusion detection, in terms of precision-recall curves. This dataset (that at the moment is limited by our ability to annotate occluded regions to a subset of the full Middlebury, but that we will continue to expand over time), as well as the implementation of our algorithm in source format will be released publicly after the anonymous review process is successfully completed.

2 Joint Occlusion Detection and Optical Flow Estimation

In this section and in appendix A, we show how the assumptions (a)-(b) can be used to formulate occlusion detection and optical flow estimation as a joint optimization problem. We assemble a functional that penalizes the (unknown) optical flow residual in the (un-known) co-visible regions, as well as the area of the occluded region. The resulting optimization problem has to be solved jointly with respect to the unknown optical flow field, and the indicator function of the occluded region.

Let $I : D \subset \mathbb{R}^2 \times \mathbb{R}^+ \to \mathbb{R}^+;\ (x,t) \mapsto I(x,t)$ be a grayscale time-varying image defined on a domain D. Under the assumptions (a)-(b), the relation between two consecutive frames in a video $\{I(x,t)\}_{t=0}^T$ is given by

$$I(x,t) = \begin{cases} I(w(x,t), t+dt) + n(x,t), & x \in D\backslash\Omega(t;dt) \\ \rho(x,t), & x \in \Omega(t;dt) \end{cases} \tag{1}$$

where $w : D \times \mathbb{R}^+ \to \mathbb{R}^2; x \mapsto w(x,t) \doteq x + v(x,t)$ is the domain deformation mapping $I(x,t)$ onto $I(x, t+dt)$ everywhere *except at occluded regions.* Usually *optical flow* denotes the incremental displacement $v(x,t) \doteq w(x,t) - x$. The occluded region Ω can change over time depending on the temporal sampling interval dt and is not necessarily simply-connected; so even if we call Ω the occluded region (singular), it is understood that it can be made of several disconnected portions. Inside Ω, the image can take any value $\rho : \Omega \times \mathbb{R}^+ \to \mathbb{R}^+$ that is in general unrelated to $I(w(x), t+dt)_{|x\in\Omega}$. In the limit $dt \to 0$, $\Omega(t;dt) = \emptyset$. Because of (almost-everywhere) continuity of the scene and its motion (i), and because the additive term $n(x,t)$ compounds the effects of a large number of independent phenomena[3] and therefore we can invoke the Law of Large Numbers (ii), in general we have that

$$\text{(i)} \quad \lim_{dt\to 0} \Omega(t;dt) = \emptyset, \quad \text{and} \quad \text{(ii)} \quad n \overset{IID}{\sim} \mathcal{N}(0,\lambda) \tag{2}$$

i.e., the additive uncertainty is normally distributed in space and time with an isotropic and small variance $\lambda > 0$. We define the residual $e : D \to \mathbb{R}$ on the entire image domain $x \in D$, via

$$\begin{aligned} e(x,t;dt) &\doteq I(x,t) - I(w(x,t), t+dt) \\ &= \begin{cases} n(x,t), & x \in D\backslash\Omega \\ \rho(x,t) - I(w(x,t), t+dt), & x \in \Omega \end{cases} \end{aligned} \tag{3}$$

[3] $n(x,t)$ collects all unmodeled phenomena including deviations from Lambertian reflection, illumination changes, quantization error, sensor noise, and later also linearization error. It does *not* capture occlusions, since those are explicitly modeled.

which we can write as the sum of two terms, $e_1 : D \to \mathbb{R}$ and $e_2 : D \to \mathbb{R}$, also defined on the entire domain D in such a way that

$$\begin{cases} e_1(x,t;dt) \doteq \rho(x,t) - I(w(x,t), t+dt), & x \in \Omega \\ e_2(x,t;dt) \doteq n(x,t), & x \in D\backslash\Omega. \end{cases} \tag{4}$$

Note that e_2 is undefined in Ω, and e_1 is undefined in $D\backslash\Omega$, in the sense that they can take any value there, including zero, which we will assume henceforth. We can then write, for any $x \in D$,

$$I(x,t) = I(w(x,t), t+dt) + e_1(x,t;dt) + e_2(x,t;dt) \tag{5}$$

and note that, because of (i) e_1 is *large but sparse*,[4] while because of (ii) e_2 is *small but dense*[4]. We will use this as an inference criterion for w, seeking to optimize a data fidelity term that minimizes the number of nonzero elements of e_1 (a proxy of the area of Ω), and the negative log-likelihood of n.

$$\begin{aligned} \psi_{\text{data}}(w, e_1) &\doteq \|e_1\|_{\mathbb{L}^0(D)} + \frac{1}{\lambda}\|e_2\|_{\mathbb{L}^2(D)} \quad \text{subject to (5)} \\ &= \frac{1}{\lambda}\|I(x,t) - I(w(x,t), t+dt) - e_1\|_{\mathbb{L}^2(D)} + \|e_1\|_{\mathbb{L}^0(D)} \end{aligned} \tag{6}$$

where $\|f\|_{\mathbb{L}^0(D)} \doteq |\{x \in D | f(x) \neq 0\}|$ and $\|f\|_{\mathbb{L}^2(D)} \doteq \int_D |f(x)|^2 dx$. Unfortunately, we do not know anything about e_1 other than the fact that it is sparse, and that what we are looking for is $\chi(\Omega) \propto e_1$, where $\chi : D \to \mathbb{R}^+$ is the characteristic function that is non-zero when $x \in \Omega$, i.e., where the occlusion residual is non-zero. So, the data fidelity term depends on w *but also on the characteristic function of the occlusion domain* Ω. For a sufficiently small dt, we can approximate[5], for any $x \in D\backslash\Omega$,

$$I(x, t+dt) = I(x,t) + \nabla I(x,t) v(x,t) + n(x,t) \tag{9}$$

where the linearization error has been incorporated into the uncertainty term $n(x,t)$. Therefore, following the same previous steps, we have

$$\psi_{\text{data}}(v, e_1) = \|\nabla I v + I_t - e_1\|_{\mathbb{L}^2(D)} + \lambda\|e_1\|_{\mathbb{L}^0(D)}. \tag{10}$$

Since we typically do not know the variance λ of the process n, we will treat it as a tuning parameter, and because ψ_{data} or $\lambda\psi_{\text{data}}$ yield the same minimizer, we have attributed the multiplier λ to the second term. In addition to the data term, because the unknown v is infinite-dimensional and the problem is ill-posed, we need to impose regularization, for instance by requiring that the total variation (TV) be small

$$\psi_{\text{reg}}(v) = \mu\|v_1\|_{TV} + \mu\|v_2\|_{TV} \tag{11}$$

where v_1 and v_2 are the first and second components of the optical flow v, μ is a multiplier factor to weight the strength of the regularizer and the weighted isotropic TV norm is defined by

$$\|f\|_{TV(D)} = \int_D \sqrt{(g_1(x)\nabla_x f(x))^2 + (g_2(x)\nabla_y f(x))^2} dx,$$

[4] *Sparse* stands for almost everywhere zero on D. Similarly, *dense* stands for almost everywhere non-zero.

[5] In a digital image, both domains D and Ω are discretized into a lattice, and dt is fixed. Therefore, spatial and temporal derivative operators are approximated, typically, by first-order differences. We use the formal notation

$$\nabla I(x,t) \doteq \begin{bmatrix} I\left(x + \begin{bmatrix} 1 \\ 0 \end{bmatrix}, t\right) - I(x,t) \\ I\left(x + \begin{bmatrix} 0 \\ 1 \end{bmatrix}, t\right) - I(x,t) \end{bmatrix}^T \tag{7}$$

$$I_t(x,t) \doteq I(x, t+dt) - I(x,t). \tag{8}$$

where g_1 and g_2 are given by

$$
\begin{aligned}
g_1(x) &= exp(-\zeta\|\nabla_x I(x)\|_2) + \nu, &(12)\\
g_2(x) &= exp(-\zeta\|\nabla_y I(x)\|_2) + \nu. &(13)
\end{aligned}
$$

where ν is small constant, preventing g_1 and g_2 to take the value 0 and ζ is a normalizing factor. TV is desirable in the context of occlusion detection because it does not penalize motion discontinuities significantly. The overall problem can then be written as the minimization of the cost functional $\psi = \psi_{\text{data}} + \psi_{\text{reg}}$, which is

$$\hat{v}_1, \hat{v}_2, \hat{e}_1 = \arg\min_{v_1, v_2, e_1} \|\nabla I v + I_t - e_1\|^2_{\mathbb{L}^2(D)} + \lambda\|e_1\|_{\mathbb{L}^0(D)} + \mu\|v_1\|_{TV(D)} + \mu\|v_2\|_{TV(D)} \tag{14}$$

In a digital image, the domain D is quantized into an $M \times N$ lattice Λ, so we can write (14) in matrix form as:

$$\hat{v}_1, \hat{v}_2, \hat{e}_1 = \arg\min_{v_1, v_2, e_1} \frac{1}{2}\|A[v_1, v_2, e_1]^T + b\|^2_{\ell_2} + \lambda\|e_1\|_{\ell_0} + \mu\|v_1\|_{TV} + \mu\|v_2\|_{TV} \tag{15}$$

where $e_1 \in \mathbb{R}^{MN}$ is the vector obtained from stacking the values of $e_1(x,t)$ on the lattice Λ on top of one another (column-wise), and similarly with the vector field components $\{v_1(x,t)\}_{x\in\Lambda}$ and $\{v_2(x,t)\}_{x\in\Lambda}$ stacked into MN-dimensional vectors $v_1, v_2 \in \mathbb{R}^{MN}$. The spatial derivative matrix A is given by

$$A = [diag(\nabla_x I) \;\; diag(\nabla_y I) \;\; -\mathcal{I}],$$

where $\mathcal{I}$ is the $MN \times MN$ identity matrix, and the temporal derivative values $\{I_t(x,t)\}_{x\in\Lambda}$ are stacked into b. For finite-dimensional vectors $u \in \mathbb{R}^{MN}$, $\|u\|_{\ell^2} = \sqrt{\langle u, u\rangle}$, $\|u\|_{\ell_0} = |\{u_i | u_i \neq 0\}|$ and $\|u\|_{TV}$ is defined as

$$\|u\|_{TV} = \sum \sqrt{((g_1)_i(u_{i+1} - u_i))^2 + ((g_2)_i(u_{i+M} - u_i))^2}$$

where g_1 and g_2 are the stacked versions of $\{g_1(x)\}_{x\in\Lambda}$ and $\{g_2(x)\}_{x\in\Lambda}$.

The problem (15) is NP-hard when solved with respect to the variable e_1 whose nonzero elements indicates the occluded region at each pixel in the image. A straightforward relaxation into a convex would simply replace the ℓ_0 norm with ℓ_1. Unfortunately, this implies that "bright" occluded regions are penalized more than "dim" ones, which is clearly not desirable. Therefore, we relax the ℓ_0 norm with the *weighted-ℓ_1* norm such that

$$\hat{v}_1, \hat{v}_2, \hat{e}_1 = \arg\min_{v_1, v_2, e_1} \frac{1}{2}\|A[v_1, v_2, \; e_1]^T + b\|^2_{\ell_2} + \lambda\|We_1\|_{\ell_1} + \mu\|v_1\|_{TV} + \mu\|v_2\|_{TV}. \tag{16}$$

where W is a diagonal matrix and resort to an iterative procedure called *reweighted-ℓ_1*, proposed by Candès et al. [9] to adapt the weights so as to better approximate the ℓ_0 norm. W is initially set to be the identity matrix, and correspondingly (16) is the customary convex relaxation[6] of the original NP-hard problem [36]. Each iteration has a globally optimal solution that can be reached efficiently from any initial condition. An improved approximation of the l_0 norm can be obtained by adapting the weight W iteratively, for instance choosing W to be a diagonal with elements $w(x) \approx 1/(|e_1(x)| + \epsilon)$ as proposed in [9]. The resulting solution of (16) greatly improves sparsity, and the residual e_1 is closer to a piecewise constant (indicator) function, as shown in Fig. 1.

Note that the residual e_1 in (5) is sometimes referred to as modeling *illumination changes* [28, 22, 35, 19]. However, even though the model (5) appears similar, the *priors* on e_1 are rather different. They favor smooth illumination changes; we favor sparse occlusions. While sparsity follows directly from the assumption (i), illumination changes would require a *reflectance function* to be modeled. Instead, all models of the form (5) lump reflectance and illumination into a single irradiance term [30].

[6]This norm has been previously used in optical flow estimation, and it makes sense in that context where occlusions are the "nuisance factors." In our context, however, occlusions are the object of inference, and we do not wish to suppress them in order to provide an optical flow reading in the occluded region, where it is undefined. Instead, optical flow is the nuisance. Therefore, while interesting, this interpretation offers no insight. Instead, we prefer using the reweighted approach as a better approximation of the original problem (16) that does not penalize bright occlusions.

3 Minimization with Nesterov's Algorithm

In this section, we describe an efficient algorithm to solve (16) based on Nesterov's first order scheme [23] which provides $O(1/k^2)$ convergence in k iterations, whereas for standard gradient descent, it is $O(1/k)$, a considerable advantage for a large scale problem such as (16). To simplify the notation we let $(e_1)_i \doteq w_i(e_1)_i$, so that $A \doteq [diag(\nabla_x I) \;\; diag(\nabla_y I) \quad -W^{-1}]$. The main steps of the algorithm are shown in the following table

Initialize v_1^0, v_2^0, e_1^0. For $k \geq 0$

1. Compute $\nabla\psi(v_1^k, v_2^k, e_1^k)$
2. Compute α_k and β_k
$$\alpha_k = 1/2(k+1), \tau_k = 2/(k+3)$$
3. Compute y_k
$$y_k = [v_1^k, v_2^k, e_1^k]^T - (1/L)\nabla\psi(v_1^k, v_2^k, e_1^k),$$
4. Compute z_k
$$z_k = [v_1^0, v_2^0, e_1^0]^T - (1/L)\sum_{i=0}^{k} \alpha_i \nabla\psi(v_1^i, v_2^i, e_1^i),$$
5. Update $[v_1^k, v_2^k, e_1^k]^T = \tau_k z_k + (1-\tau_k)y_k$.

Stop when the solution converges.

In order to implement this scheme, we need to address the nonsmooth nature of ℓ_1. This is done in [24], that has already been used profitably for noise reduction, inpainting and deblurring [42, 12], and incorporated in software libraries for sparse recovery [4]. In our case, we write $\psi(v_1, v_2, e_1)$ as a summation of terms

$$\psi(v_1, v_2, e_1) = \psi_1(v_1, v_2, e_1) + \lambda\psi_2(e_1) + \mu\psi_3(v_1) + \mu\psi_4(v_2),$$

and compute the gradient of each term separately: The first is straightforward

$$\nabla_{v_1,v_2,e_1}\psi_1(v_1, v_2, e_1) = A^T A[v_1, v_2, e_1]^T + A^T b.$$

The other three, however, require smoothing. $\psi_2(e_1) = \|e_1\|_{\ell_1}$ can be rewritten in terms of its conjugate $\psi_2(e_1) = \max_{\|u\|_\infty \leq 1} \langle u, e_1\rangle$. The smooth approximation proposed in [24] is

$$\psi_2^\sigma(e_1) = \max_{\|u\|_\infty \leq 1} \langle u, e_1\rangle - \frac{1}{2}\sigma\|u\|_{\ell_2}^2 \tag{17}$$

which is differentiable; its gradient is u^σ, the optimal solution of (17). Consequently, $\nabla_{e_1}\psi_2^\sigma(e_1))$ is given by

$$u_i^\sigma = \begin{cases} \sigma^{-1}(e_1)_i, & |(e_1)_i| < \sigma, \\ \text{sgn}((e_1)_i), & \text{otherwise.} \end{cases} \tag{18}$$

Following the lines of [4], $\nabla_{v_1}\psi_3$ is given by is given by

$$\nabla_{v_1}\psi_3^\sigma(v_1) = G^T u^\sigma \tag{19}$$

where $G = [G_1, G_2]^T$, G_1 and G_2 are weighted horizontal and vertical differentiation operators, and u^σ has the form $[u^1, u^2]$ where

$$u_i^{1,2} = \begin{cases} \sigma^{-1}(G_{1,2}v_1)_i, & \|[(G_1 v_1)_i \;\; (G_2 v_1)_i]^T\|_{\ell_2} < \sigma, \\ \|[(G_1 v_1)_i \;\; (G_2 v_1)_i]^T\|_{\ell_2}^{-1}(G_{1,2}v_1)_i, & \text{otherwise.} \end{cases} \tag{20}$$

$\nabla_{v_2}\psi_4$ can also be computed in the same way. We now have all the terms necessary to compute

$$\nabla\psi(v_1, v_2, e_1) = \nabla\psi_1 + [\lambda\nabla_{e_1}\psi_2, \mu\nabla_{v_1}\psi_3, \mu\nabla_{v_2}\psi_4]^T. \tag{21}$$

We also need the Lipschitz constant L to compute the auxiliary variables y_k and z_k to minimize ψ. Since $\|G^T G\|_2$ is bounded above [12] by 8, given the coefficients λ and μ, L is given by

$$L = \max(\lambda, 8\mu)/\sigma + \|A^T A\|_2.$$

A crucial element of the scheme is the selection of σ. It trades off accuracy and speed of convergence. A large σ yields a smooth solution, which is undesirable when minimizing the ℓ_1 norm. A small σ causes slow convergence. We have chosen σ empirically, although the continuation algorithm proposed in [4] could be employed to adapt σ during convergence.

4 Minimization with Split-Bregman

We also describe an alternative method for solving the optimization problem (16) based on the split-Bregman method proposed by Goldstein and Osher [14], by decoupling the differentiable and non-differentiable portions of the cost function (16).

We replace $G_x v_{(1,2)}$ by $d_x^{(1,2)}$ and $G_y v_{(1,2)}$ by $d_y^{(1,2)}$ yielding to a constrained problem,

$$\begin{aligned}
\hat{v}_1, \hat{v}_2, \hat{e}_1 = &\operatorname*{argmin}_{v_1, v_2, e_1} \frac{1}{2\mu}\|A[v_1,\ v_2,\ e_1]^T + I_t\|_{\ell_2}^2 \\
&+ \frac{\lambda}{\mu}\|We_1\|_{\ell_1} \\
&+ \|(d_x^1, d_y^1)\|_{\ell_1} + \|(d_x^2, d_y^2)\|_{\ell_1} \\
\text{subject to:}& \\
&d_x^1 = G_x v_1,\ d_y^1 = G_y v_1, \\
&d_x^2 = G_x v_2,\ d_y^2 = G_y v_2.
\end{aligned} \tag{22}$$

where for finite dimensional vectors $u_1, u_2 \in \mathbb{R}^{MN}$, $\|(u_1, u_2)\|_{\ell_1} = \sum_{i=1}^{MN} \sqrt{(u_1)_i^2 + (u_2)_i^2}$. By relaxing the hard constraints, the cost function (22) takes a form that can be minimized by split-Bregman such that

$$\begin{aligned}
&\hat{v}_1, \hat{v}_2, \hat{e}_1, \hat{d}_{(x,y)}^{(1,2)} = \\
&\operatorname*{argmin}_{v_1, v_2, e_1, d_{(x,y)}^{(1,2)}} \frac{1}{2\mu}\|A[v_1,\ v_2,\ e_1]^T + I_t\|_{\ell_2}^2 \\
&\quad + \frac{\lambda}{\mu}\|We_1\|_{\ell_1} + \|(d_x^1, d_y^1)\|_{\ell_1} + \|(d_x^2, d_y^2)\|_{\ell_1} \\
&\quad + \frac{\beta}{2}\|d_x^1 - G_x v_1 - b_x^1\|_{\ell_2}^2 + \frac{\beta}{2}\|d_y^1 - G_y v_1 - b_y^1\|_{\ell_2}^2 \\
&\quad + \frac{\beta}{2}\|d_x^2 - G_x v_2 - b_x^2\|_{\ell_2}^2 + \frac{\beta}{2}\|d_y^2 - G_y v_2 - b_y^2\|_{\ell_2}^2
\end{aligned} \tag{23}$$

where β indicates the amount of relaxation. To solve (23), we divide the optimization problem into three subproblems and solve them iteratively. The first subproblem is

$$\begin{aligned}
\hat{v}_1^{k+1}, \hat{v}_2^{k+1} = &\operatorname*{argmin}_{v_1, v_2} \frac{1}{2\mu}\|A[v_1,\ v_2,\ e_1^k]^T + I_t\|_{\ell_2}^2 \\
&+ \frac{\beta}{2}\|(d_x^1)^k - G_x v_1 - (b_x^1)^k\|_{\ell_2}^2 + \frac{\beta}{2}\|(d_y^1)^k - G_y v_1 - (b_y^1)^k\|_{\ell_2}^2 \\
&+ \frac{\beta}{2}\|(d_x^2)^k - G_x v_2 - (b_x^2)^k\|_{\ell_2}^2 + \frac{\beta}{2}\|(d_y^2)^k - G_y v_2 - (b_y^2)^k\|_{\ell_2}^2
\end{aligned} \tag{24}$$

The solution of this problem is straightforward. From the optimality conditions, we reach to the following system of equations

$$
\begin{aligned}
&\Big(\frac{1}{\mu} I_x^T I_x + \beta(Gx^T G_x + G_y^T G_y)\Big) v_1 + \frac{1}{\mu} I_x^T I_y v_2 = \\
&\quad -\frac{1}{\mu} I_x^T(-e_1 + I_t) + \beta\Big(G_x^T(d_x^1 - b_x^1) + G_y^T(d_y^1 - b_y^1)\Big), \\
&\Big(\frac{1}{\mu} I_y^T I_y + \beta(G_x^T G_x + G_y^T G_y)\Big) v_2 + \frac{1}{\mu} I_y^T I_x v_1 = \\
&\quad -\frac{1}{\mu} I_y^T(-e_1 + I_t) + \beta\Big(G_x^T(d_x^2 - b_x^2) + G_y^T(d_y^2 - b_y^2)\Big).
\end{aligned}
$$

where to simplify the notation we have defined the diagonal matrices $I_x = diag(\nabla_x I)$ and $I_y = diag(\nabla_x I)$. Following [14], to achieve efficiency, we solve the system of equations using Gauss-Seidel's method. The component-wise Gauss-Seidel solution to this problem is given by

$$
\begin{aligned}
(v_1)_i =& \frac{-\mu(k_2)_i(I_y)_i(I_x)_i + (k_1)_i(\mu(I_y)_i^2 + (k_3)_i))}{(k_3)_i\big(\mu(I_x)_i^2 + \mu(I_y)_i^2 + (k_3)_i\big)} = G_i^1 \\
(v_2)_i =& \frac{-\mu(k_1)_i(I_y)_i(I_x)_i + (k_2)_i(\mu(I_x)_i^2 + (k_3)_i))}{(k_3)_i\big(\mu(I_x)_i^2 + \mu(I_y)_i^2 + (k_3)_i\big)} = G_i^2
\end{aligned}
\tag{25}
$$

where k_1, k_2 and k_3 are given by

$$
\begin{aligned}
(k_1)_i =& -\mu(I_x)_i(-(e_1)_i + (I_t)_i) \\
&+ \beta\Big((G_x^T G_x v_1)_i + (G_y^T G_y v_1)_i\Big) \\
&+ \beta\Big((G_x^T d_x^1)_i + (G_y^T d_y^1)_i + (G_x^T b_x^1)_i + (G_y^T b_y^1)_i\Big) \\
(k_2)_i =& -\mu(I_y)_i(-(e_1)_i + (I_t)_i) \\
&+ \beta\Big((G_x^T G_x v_2)_i + \beta(G_y^T G_y v_2)_i\Big) \\
&+ \beta\Big((G_x^T d_x^2)_i + (G_y^T d_y^2)_i + (G_x^T b_x^2)_i + (G_y^T b_y^2)_i\Big) \\
k_3 =& \beta \; diag(G_x^T G_x + G_y^T G_y).
\end{aligned}
$$

Subsequently, we need to solve the second subproblem which is given by

$$
\begin{aligned}
&\Big(\hat{d}_x^{(1,2)}\Big)^{k+1}, \Big(\hat{d}_y^{(1,2)}\Big)^{k+1} = \\
&\quad \underset{d_x^{(1,2)}, d_y^{(1,2)}}{\operatorname{argmin}} \|(d_x^{(1,2)}, d_y^{(1,2)})\|_{\ell_1} \\
&\qquad + \frac{\beta}{2}\|(d_x^{(1,2)}) - G_x v_{(1,2)} - (b_x^{(1,2)})^k\|_{\ell_2}^2 \\
&\qquad + \frac{\beta}{2}\|(d_y^{(1,2)}) - G_y v_{(1,2)} - (b_y^{(1,2)})^k\|_{\ell_2}^2.
\end{aligned}
\tag{26}
$$

This problem can be solved analytically using the generalized shrinkage formula [39] such that

$$
\begin{aligned}
\Big(\hat{d}_x^{(1,2)}\Big)^{k+1} &= \max(s^k - 1/\beta, 0)\frac{G_x v_{(1,2)}^k + (b_x^{(1,2)})^k}{s^k} \\
\Big(\hat{d}_y^{(1,2)}\Big)^{k+1} &= \max(s^k - 1/\beta, 0)\frac{G_y v_{(1,2)}^k + (b_y^{(1,2)})^k}{s^k}
\end{aligned}
\tag{27}
$$

where $s^k_{(1,2)}$ is given by

$$(s^k)_i = \sqrt{|G_x v^k_{(1,2)} + (b_x^{(1,2)})^k|^2 + |G_y v^k_{(1,2)} + (b_y^{(1,2)})^k|^2}.$$

The remaining subproblem is

$$\hat{e}_1 = \underset{e_1}{\operatorname{argmin}} \frac{1}{2\mu} \|A[v_1,\ v_2,\ e_1]^T + I_t\|^2_{\ell_2} + \frac{\lambda}{\mu} \|We_1\|_{\ell_1}. \tag{28}$$

and can also be solved using shrinkage operator. The solution is

$$(e_1)_i^{k+1} = \frac{r_i^k}{|r_i^k|} \max(|r_i^k| - \lambda w_i, 0). \tag{29}$$

where $r^k = I_x v_1^k + I_y v_2^k + I_t$.

The main steps of the algorithm can be summarized as follows

Initialize $v_1, v_2, e_1, d_x^1, d_y^1, d_x^2$ and d_y^2 with 0. For $k \geq 0$

$$v_1^{k+1} = G_1^k,\ v_2^{k+1} = G_2^k$$

$$\left(d_x^{(1,2)}\right)^{k+1} = \max(s^k - 1/\beta, 0) \frac{G_x v^k_{(1,2)} + (b_x^{(1,2)})^k}{s^k}$$

$$\left(d_x^{(1,2)}\right)^{k+1} = \max(s^k - 1/\beta, 0) \frac{G_y v^k_{(1,2)} + (b_y^{(1,2)})^k}{s^k}$$

$$\left(b_x^{(1,2)}\right)^{k+1} = \left(b_x^{(1,2)}\right)^k + \left(G_x v^{k+1}_{(1,2)} - \left(d_x^{(1,2)}\right)^{k+1}\right)$$

$$\left(b_y^{(1,2)}\right)^{k+1} = \left(b_y^{(1,2)}\right)^k + \left(G_y v^{k+1}_{(1,2)} - \left(d_y^{(1,2)}\right)^{k+1}\right)$$

$$(e_1)_i^{k+1} = \frac{r_i^k}{|r_i^k|} \max(|r_i^k| - \lambda w_i, 0)$$

Stop when the solution converges.

5 Experiments

Following Section 1.2, evaluation of our algorithm on standard datasets is not straightforward, because these typically do not provide ground-truth occlusions. The only benchmark that provides occlusion, in at least parts of the dataset, is [3], so we used it as a starting point, and generated occlusion maps as follows: for each training sequence, we computed the residual given the ground truth motion and marked the regions where the residual is high. Next, we annotated the regions where ground truth is not defined. Finally, we manually fixed obvious errors in the occlusion maps. In this section, we evaluate the motion estimation and occlusion detection performance of our approach on this dataset and on the well-known Flower Garden sequence. We have also compared our algorithm to [41], [6] and [20] quantitatively.

To handle the large motion, we run our method on a Gaussian pyramid with a scale factor 0.5 up to 5 levels. We also apply 5 warping steps at each pyramid level. In all the experiments, the coefficient λ is fixed at 0.01 while μ is increased gradually from 0.00008 to 0.01 with each warping step at each pyramid level. Relying less on the prior of the flow field at the early warping steps results in more accurate flow estimates. For the re-weighting step, we have also fixed the coefficient ϵ to 0.001. In our experiments, we also use a non-linear pre-filtering of the images to reduce the influence of illumination changes [27, 41, 32] to initialize the re-weighting stage with an accurate flow field. However, at re-weighting steps we use the original images since pre-filtering reduces the occlusion detection accuracy.

We start with unit weights $W = \mathcal{I}$ and solve the convex problem (16) (referred as Huber-ℓ_1 model in our experiments). We then adapt the weights iteratively, thus improving sparsity and achieving a better approximation of the indicator function e_1 of the occluded domain, Fig. 1. One can also observe a gradual improvement of the sparsity of $|We|$ after each re-weighting iteration, Fig. 2. At each step, the accuracy of occlusion detection also improves.

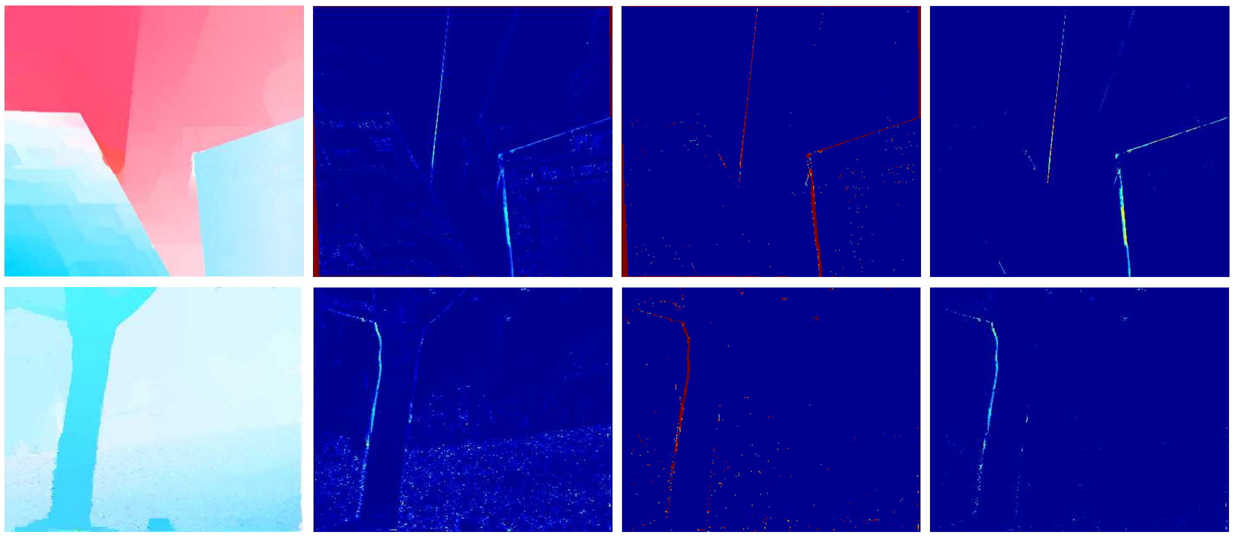


Figure 1: *The result of the proposed approach on "Venus" from [3] and "Flower Garden." The first column shows the motion estimates, color-coded as in [3], the second is the residual $I(x,t) - I(w(x), t+dt)$ before re-weighting stage; the third shows $|We_1|$ after re-weighting, and the fourth is the sparse error term e_1.*

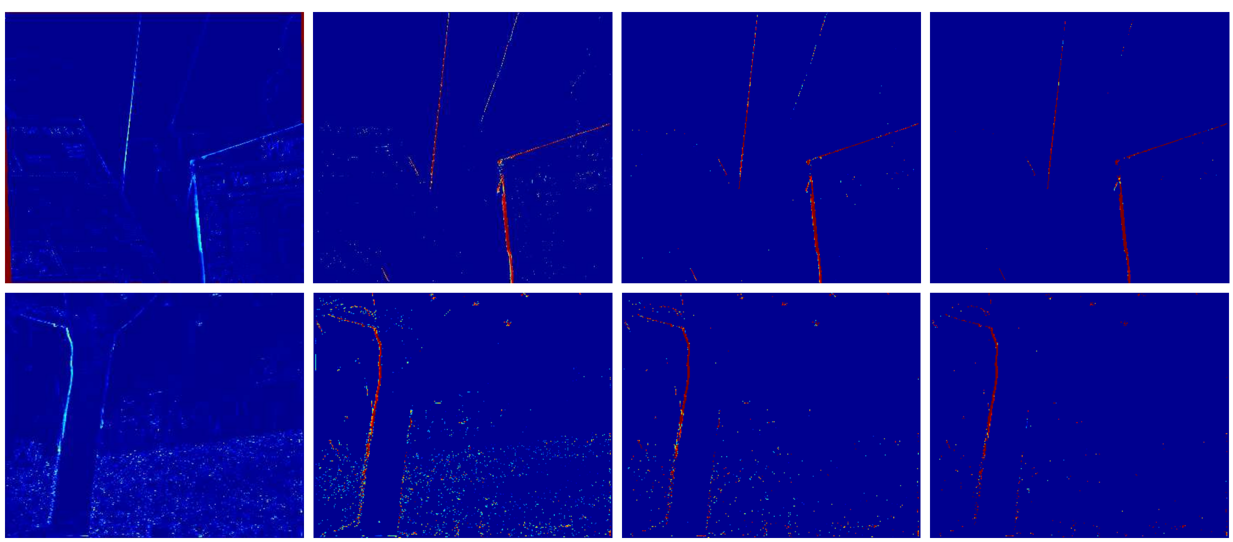


Figure 2: *This figure illustrates the initial estimate of the error term e_1 (first column) and how sparsity of $|We_1|$ improves with each of the three re-weighting iterations.*

Representative results for the Flower Garden sequence are shown in Fig. 3, where the complex occlusions produced by the foliage are also detected successfully.

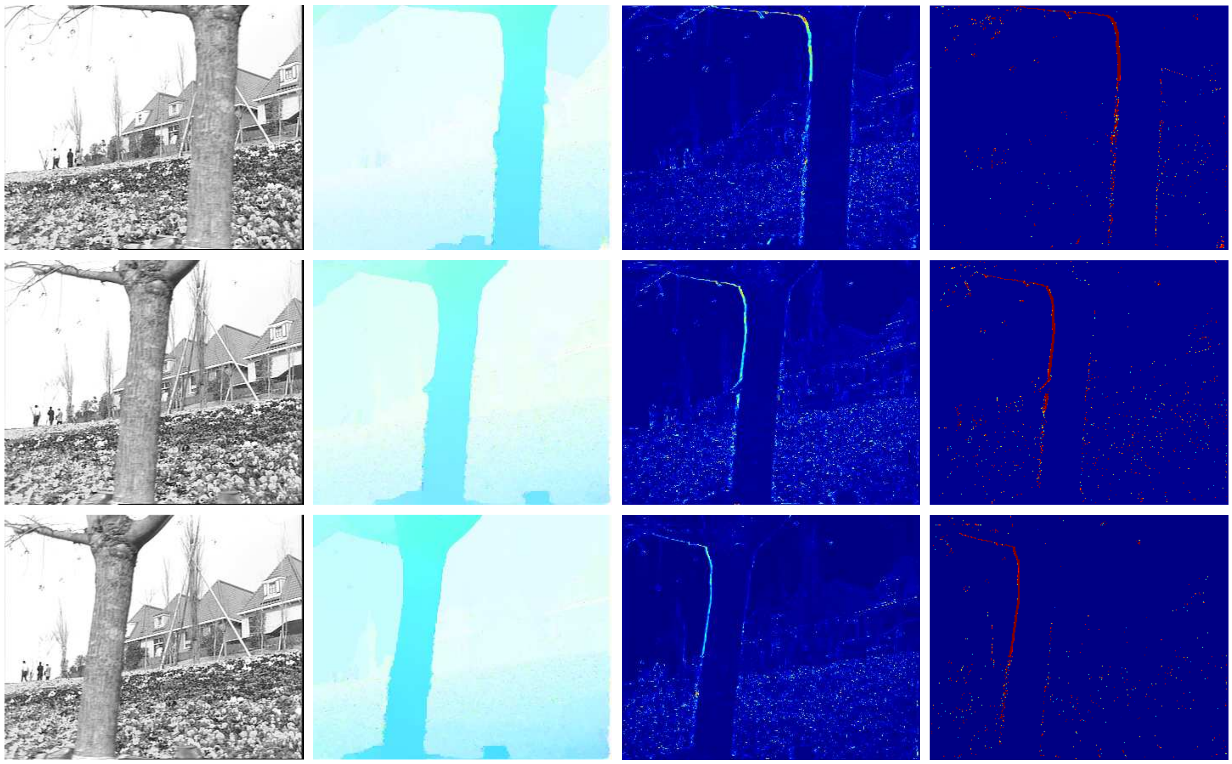
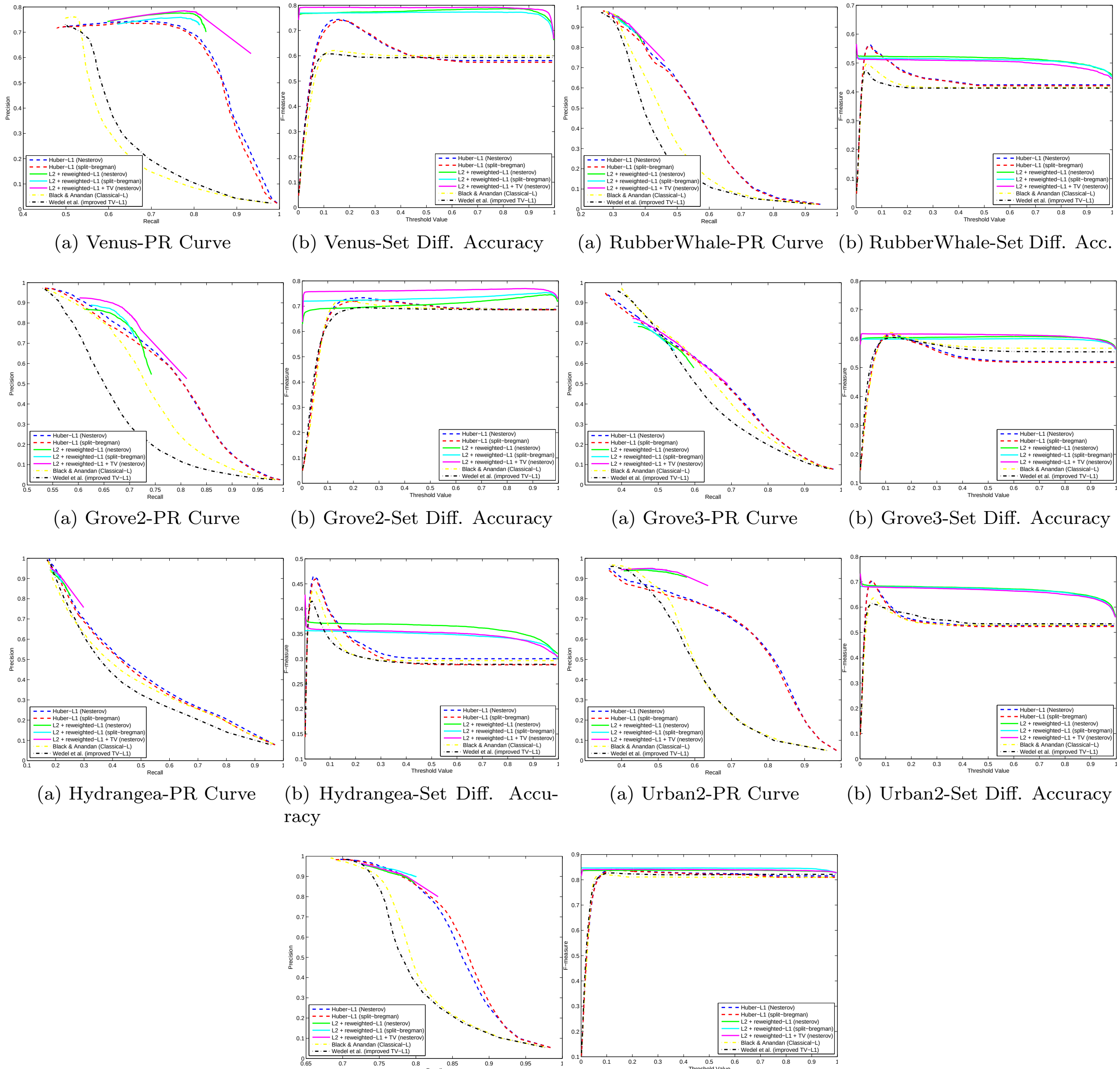


Figure 3: *Occlusion and motion estimates for more frames of the Flower Garden. Left to right: initial frame, flow estimate (left), initial estimate of the error term e_1 (middle), and occluded region (right).*

In Fig. 5 and Fig. 6 we show the effects of re-weighting on the Middlebury data set. The weighted e_1 is not only sparser compared to the residual $|I(x,t) - I(w(x), t+dt)|$ computed before the re-weighting steps but also has a superior occlusion detection accuracy unlike the residual which contains regions that are not occluded. One might think that the residual could just be thresholded, instead of iteratively re-weighted. To evaluate that, we have generated precision-recall curves and observed the change of occlusion detection performance in terms of F-measure by thresholding both signals while varying the threshold value in the interval $[0, 1]$, Fig. 4. In most cases, the accuracy of the re-weighting approach is superior and more stable under the varying threshold values since $|We_1|$ better approximates an indicator function. Therefore, one can just choose non-zero elements of e_1[7] to detect occluded regions instead of searching for a global threshold. Note that here the weight matrix W is the one computed at the previous re-weighting step. We have also observed that the precision-recall curves for $|We_1|$ does not span the whole recall range, since recall value 1 is not reachable unless all the zero-elements added to the decision which is not meaningful for the analysis of a sparse signal (Fig. 4, PR-curves). We have also compared our approach to the robust flow estimation methods proposed by Black and Anandan [6], using the improved version by Sun et al. [32](Classical-L), and Wedel et al. [41] by evaluating the occlusion detection accuracy on the residual $|I(x,t) - I(w(x), t+dt)|$ computed using their flow fields.

One might be tempted to regularize the geometry of the occluded region, for instance by adding a regularizing term $\|We_1\|_{TV}$ to (16). We have also evaluated this model, Fig. 4. However, occlusions can manifest themselves with very complex geometry and topology, as the Hydrangea in Fig. 6 and Fig. 4 illustrate. In such cases, a geometric regularizer is counter-productive as it generates a large number of

[7] Notice that $w(x) > 0$, $\forall x$. Therefore, $We_1 \neq 0 \iff e_1 \neq 0$



(a) Venus-PR Curve (b) Venus-Set Diff. Accuracy (a) RubberWhale-PR Curve (b) RubberWhale-Set Diff. Acc.

(a) Grove2-PR Curve (b) Grove2-Set Diff. Accuracy (a) Grove3-PR Curve (b) Grove3-Set Diff. Accuracy

(a) Hydrangea-PR Curve (b) Hydrangea-Set Diff. Accuracy (a) Urban2-PR Curve (b) Urban2-Set Diff. Accuracy

(a) Urban3-PR Curve (b) Urban3-Set Diff. Accuracy

Figure 4: *Comparison of occlusion detection accuracy with [6] and [41] in terms of precision-recall curves and F-measure.*

missed detections.

We have compared our occlusion detection results to [20], using the code provided on-line by the authors. Table 1 shows that we outperform [20]. Comparing motion estimates gives an unfair advantage to our algorithm because their approach is based on quantized disparity values, so the accuracy of our motion estimates is predictably superior.

We have also compared the accuracy of the solution of Nesterov's algorithm and split-Bregman's method

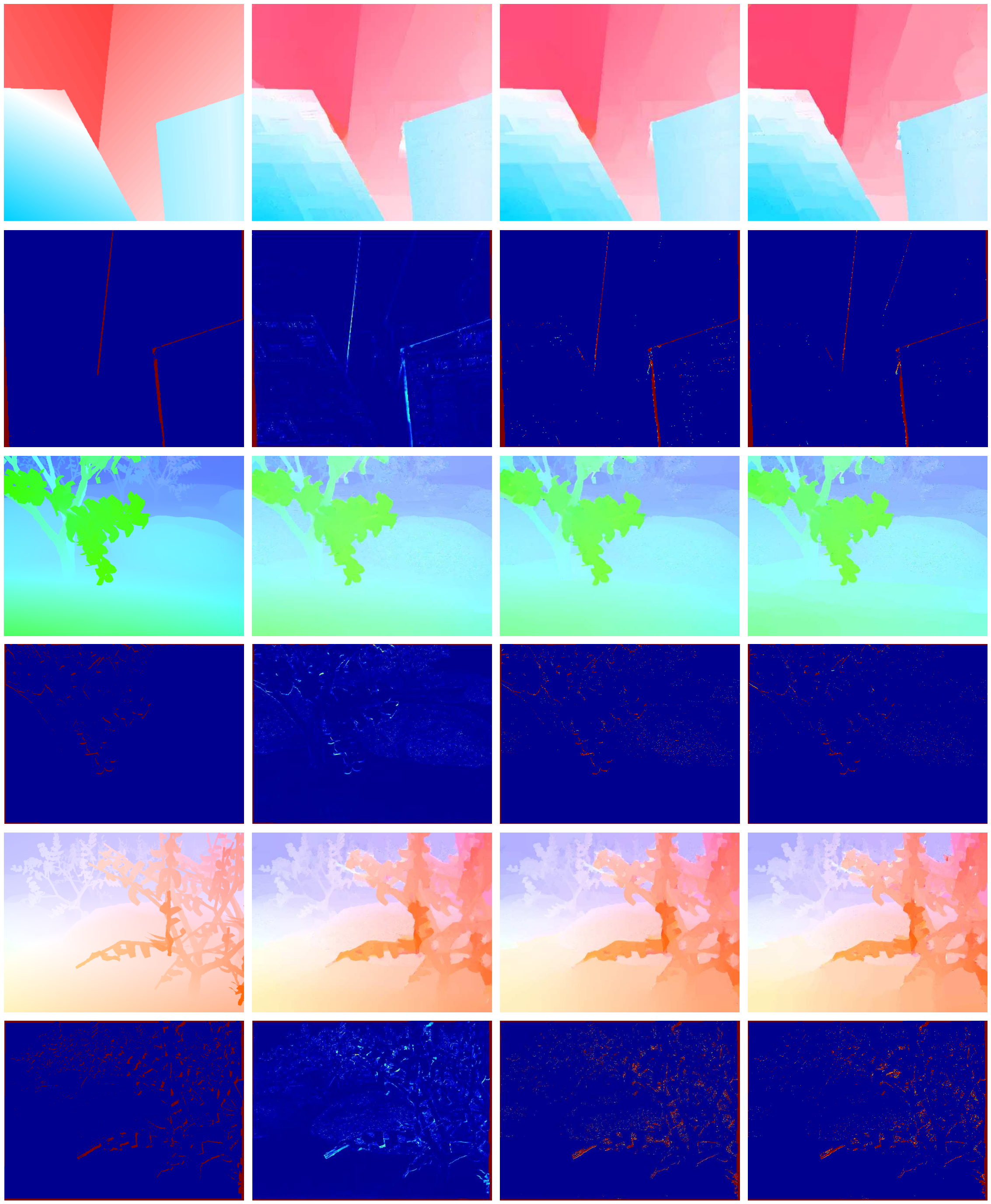


Figure 5: *This figure presents the occlusion and motion estimates on the sequences Venus, Grove2 and Grove3 from Middlebury dataset. Each sequence occupies two rows. The odd rows, left to right: ground truth optical flow, flow estimates before re-weighting stage and flow estimates after reweighing with Nesterov's algorithm and split-Bregman method. The even rows, left to right: ground truth occluded regions, the initial estimate of the error term* e_1*, the estimate of* $|We_1|$ *after the reweighting step with Nesterov's algorithm and split-Bregman method.*

	Venus	RubberWhale	Hydrangea	Grove2	Grove3	Urban2	Urban3
F-measure [20]	0.63	0.28	0.31	0.62	0.52	0.43	0.53
F-measure (our method)	**0.77**	**0.52**	**0.37**	**0.67**	**0.60**	**0.69**	**0.83**

Table 1: *A comparison of the F-measure of our algorithm and [20] on the Middlebury dataset. Since Kolmogorov et al. [20] provide an occlusion detector whose output is binary, we simply compute the precision and recall of the output and report the F-measure based on these values. For comparison, we chose non zero elements of e_1 as detected occlusions and provide F-measure with respect to them.*

and their convergence speed, Fig. 5, Fig. 6, Fig. 4, Table 2 and Table 3. Both methods provide similar performance both in occlusion detection and motion estimation. However, split-Bregman method converges significantly faster, Table 2.

We have evaluated the accuracy of the flow estimates of our method and compared to other robust flow estimation techniques [6, 32, 41], Table 3. Huber$-\ell_1-$TV model minimized with Nesterov's algorithm provides superior accuracy. However, once the re-weighting stage is initialized with these estimates, and flow estimation is performed on the original images instead of the pre-filtered ones, the accuracy decreases.

	Venus	RubberWhale	Hydrangea	Grove2	Grove3	Urban2	Urban3
Nesterov's algorithm	222 secs	342 secs	355 secs	463 secs	494 secs	499 secs	483 secs
Split Bregman method	90 secs	111 secs	133 secs	260 secs	360 secs	288 secs	277 secs

Table 2: *The comparison of convergence time of the split-Bregman method and Nesterov's algorithm.*

6 Discussion

We have presented an algorithm to detect occlusions and establish correspondence between two images. It leverages on a formulation that, starting from standard assumptions (Lambertian reflection, constant diffuse illumination), arrives at a variational optimization problem. We have shown how this problem can be relaxed into a sequence of convex optimization schemes, each having a globally optimal solution, and presented two efficient numerical schemes for solving it.

We emphasize that our approach does *not* assume a rigid scene, or a single moving object. It also does *not* assume that the occluded region is simply connected. Instead, our model is general under the assumptions (a)-(b) as we show in Appendix A, and allows arbitrary (piece-wise diffeomorphic) domain deformations, corresponding to an arbitrary number of moving or deforming objects, and an arbitrary number of simply connected occluded regions (jointly represented by a multiply-connected domain Ω).

The fact that occlusion detection reduces to a two-phase segmentation of the domain into occluded

	Venus	RubberWhale	Hydrangea	Grove2	Grove3	Urban2	Urban3
Huber$-\ell_1-$TV (Nesterov)	3.99/0.28	2.94/0.09	2.09/0.17	2.19/0.15	6.78/ 0.67	2.59/0.29	4.35/0.66
ℓ_2-reweighted$-\ell_1-$TV (Nesterov)	3.96/0.31	5.09/0.16	2.36/0.19	2.60/0.17	7.71/0.78	3.41/0.38	4.91/0.76
ℓ_2-reweighted$-\ell_1-$TV (split-Bregman)	4.09/0.33	4.91/0.16	2.34/0.19	2.31/0.15	7.72/0.75	2.86/0.35	4.20/0.63
Black & Anandan[6]	7.81/0.44	5.06/0.14	2.48/0.21	2.76/0.20	6.90/0.75	4.06/0.54	11.18/0.94
Classic-L [32]	4.75/0.29	3.15/0.09	2.06/0.17	2.49/0.17	6.49/0.66	2.96/0.37	4.72/0.60
Wedel et al. (Improved L1-TV) [41]	4.45/0.30	3.61/0.11	2.25/0.18	3.26/0.23	7.07/0.69	2.74/0.36	6.26/0.64

Table 3: *Quantitative comparison of the proposed models and other robust flow estimation methods [6, 32, 41] in terms of Average Angular Error (AAE) / Average End Point Error (AEPE)*

Ω and visible region $D \backslash \Omega$ should not confuse the reader familiar with the *image segmentation* literature whereby two-phase segmentation of *one object* (foreground) from the background can be posed as a convex optimization problem [11]. Note that in the approach of [11] the problem can be made convex only in occluded region term, e_1, but not jointly in both e_1 and the motion field, v. Therefore, such an approach does not in general yield a global minimum.

The limitations of our approach stand mostly in its dependency from the regularization coefficients λ, μ and coefficient σ in the optimization. In the absence of some estimate of the variance coefficient λ, one is left with painstakingly tuning it by trial-and-error. Similarly, μ is a parameter that, like in any classification problem, trades off missed detections and false alarms, and therefore no single value is "optimal" in any meaningful sense. These limitations are shared by most variational optical flow estimation algorithms.

References

[1] L. Alvarez, R. Deriche, T. Papadopoulo, and J. Sánchez. Symmetrical dense optical flow estimation with occlusions detection. *International Journal of Computer Vision*, 75(3):371–385, 2007.

[2] A. Ayvaci, M. Raptis, and S. Soatto. Occlusion detection and motion estimation with convex optimization. In *Advances in Neural Information Processing Systems*. 2010.

[3] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. Black, and R. Szeliski. A database and evaluation methodology for optical flow. In *Proc. of the International Conference on Computer Vision*, volume 5, 2007.

[4] S. Becker, J. Bobin, and E. Candes. Nesta: A fast and accurate first-order method for sparse recovery. *Arxiv preprint arXiv*, 904, 2009.

[5] R. Ben-Ari and N. Sochen. Variational stereo vision with sharp discontinuities and occlusion handling. In *Proc. of Internation Conference on Computer Vision*, pages 1–7, 2007.

[6] M. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63(1):75–104, 1996.

[7] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *Proc. of European Conference on Computer Vision*, pages 25–36, 2004.

[8] A. Bruhn, J. Weickert, and C. Schnörr. Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods. *International Journal of Computer Vision*, 61(3):211–231, 2005.

[9] E. Candes, M. Wakin, and S. Boyd. Enhancing sparsity by reweighted L1 minimization. *Journal of Fourier Analysis and Applications*, 14(5):877–905, 2008.

[10] V. Caselles, B. Coll, and J.-M. Morel. Topographic maps and local contrast changes in natural images. *International Journal of Computer Vision*, 33(1):5–27, 1999.

[11] T. Chan, S. Esedoglu, and M. Nikolova. Algorithms for finding global minimizers of denoising and segmentation models. *SIAM Journal on Applied Mathematics*, 66(1632-1648):1, 2006.

[12] J. Dahl, P. Hansen, S. Jensen, and T. Jensen. Algorithms and software for total variation image reconstruction via first-order methods. *Numerical Algorithms*, pages 67–92, 2009.

[13] J. J. Gibson. *The ecological approach to visual perception.* LEA, 1984.

[14] T. Goldstein and S. Osher. The split Bregman method for L1 regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.

[15] B. Horn and B. Schunck. Determining optical flow. *Computer Vision*, 17:185–203, 1981.

[16] S. Ince and J. Konrad. Occlusion-aware optical flow estimation. *IEEE Transactions on Image Processing*, 17(8):1443–1451, 2008.

[17] J. Jackson, A. J. Yezzi, and S. Soatto. Dynamic shape and appearance modeling via moving and deforming layers. *International Journal of Computer Vision*, 2008.

[18] J. D. Jackson, A. J. Yezzi, and S. Soatto. Dynamic shape and appearance modeling via moving and deforming layers. In *Proc. of Workshop on Energy Minimization in Computer Vision and Pattern Recognition (EMMCVPR)*, pages 427–438, 2005.

[19] Y. Kim, A. Martínez, and A. Kak. Robust motion estimation under varying illumination. *Image and Vision Computing*, 23(4):365–375, 2005.

[20] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions via graph cuts. In *Proc. of International Conference on Computer Vision*, volume 2, pages 508–515, 2001.

[21] K. Lim, A. Das, and M. Chong. Estimation of occlusion and dense motion fields in a bidirectional Bayesian framework. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 712–718, 2002.

[22] S. Negahdaripour. Revised definition of optical flow: Integration of radiometric and geometric cues for dynamic scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 961–979, 1998.

[23] Y. Nesterov. A method for unconstrained convex minimization problem with the rate of convergence O $(1/k^2)$. In *Doklady AN SSSR*, volume 269, pages 543–547, 1983.

[24] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1):127–152, 2005.

[25] M. Proesmans, L. Van Gool, and A. Oosterlinck. Determination of optical flow and its discontinuities using a non-linear diffusion. In *Proc. of European Conference of Computer Vision*, 1994.

[26] C. P. Robert. *The Bayesian Choice*. Springer Verlag, New York, 2001.

[27] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[28] D. Shulman and J. Herve. Regularization of discontinuous flow fields. In *Proc. of Workshop on Visual Motion*, pages 81–86, 1989.

[29] S. Soatto and A. Yezzi. Deformotion: deforming motion, shape average and the joint segmentation and registration of images. In *Proc. of the European Conference on Computer Vision*, volume 3, pages 32–47, 2002.

[30] S. Soatto, A. J. Yezzi, and H. Jin. Tales of shape and radiance in multiview stereo. In *Proc. of Internation Conference on Computer Vision*, pages 974–981, October 2003.

[31] C. Strecha, R. Fransens, and L. Van Gool. A probabilistic approach to large displacement optical flow and occlusion detection. In *ECCV Workshop SMVP*, pages 71–82. Springer, 2004.

[32] D. Sun, S. Roth, and M. Black. Secrets of optical flow estimation and their principles. In *Proc. of Conference on Computer Vision and Pattern Recognition*, pages 2432–2439, 2010.

[33] J. Sun, Y. Li, S. Kang, and H. Shum. Symmetric stereo matching for occlusion handling. In *Proc. of Conference on Computer Vision and Pattern Recognition*, volume 2, page 399, 2005.

[34] G. Sundaramoorthi, P. Petersen, V. S. Varadarajan, and S. Soatto. On the set of images modulo viewpoint and contrast changes. In *Proc. of Conference on Computer Vision and Pattern Recognition*, 2009.

[35] C. Teng, S. Lai, Y. Chen, and W. Hsu. Accurate optical flow computation under non-uniform brightness variations. *Computer Vision and Image Understanding*, 97(3):315–346, 2005.

[36] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996.

[37] A. Verri and T. Poggio. Motion field and optical flow: Qualitative properties. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(5):490–498, 1989.

[38] J. Wang and E. Adelson. Representing moving images with layers. *IEEE Transactions on Image Processing*, 3(5):625 –638, 1994.

[39] Y. Wang, W. Yin, and Y. Zhang. A fast algorithm for image deblurring with total variation regularization. Technical report, CAAM Technical Reports, Rice University, 2007.

[40] A. Wedel, D. Cremers, T. Pock, and H. Bischof. Structure- and motion-adaptive regularization for high accuracy optic flow. In *Proc. of International Conference on Computer Vision*, 2009.

[41] A. Wedel, T. Pock, C. Zach, H. Bischof, and D. Cremers. An improved algorithm for TV-L1 optical flow. In *Proc. of Statistical and Geometrical Approaches to Visual Motion Analysis: International Dagstuhl Seminar*, 2008.

[42] P. Weiss, L. Blanc-Féraud, and G. Aubert. Efficient Schemes for Total Variation Minimization Under Constraints in Image Processing. *SIAM Journal on Scientific Computing*, 31:2047, 2009.

[43] M. Werlberger, W. Trobin, T. Pock, A. Wedel, D. Cremers, and H. Bischof. Anisotropic Huber-L1 Optical Flow. In *Proc. of British Machine Vision Conference*, 2009.

[44] J. Xiao, H. Cheng, H. Sawhney, C. Rao, M. Isnardi, et al. Bilateral filtering-based optical flow estimation with occlusion detection. In *Proc. of European Conference of Computer Vision*, volume 3951, page 211, 2006.

A Ambient-Lambert model

In this section we show how to go from the assumptions (a)-(c) in section 1 to eq. (6). Let the scene $\{S, \rho\}$ be described by shape $S \subset \mathbb{R}^3$ (a collection of piece-wise smooth surfaces) and reflectance $\rho : S :\to \mathbb{R}^k$ (diffuse albedo). Deviations from diffuse reflectance will not be modeled explicitly and lumped as error (inter-reflection, sub-surface scattering, specular reflection, cast shadows). Coarse illumination changes are modeled as a contrast transformation of the image range, and all other illumination effects are lumped into the additive error. The large number of independent phenomena being aggregated into such an error make it suitable to be modeled as a Gaussian random process (eq. (2)-ii). Under these assumptions, the radiance ρ emitted by an area element around a point $p \in S$ is modulated by a monotonic continuous transformation m to yield the irradiance I measured at a pixel element x, except for the discrepancy $n : D \to \mathbb{R}^k_+$, and the correspondence between the point $p \in S$ and the pixel $x \in D$ is due to the motion of the viewer $g \in SE(3)$, the special Euclidean group of rotations and translations in three dimensional (3-D) space:

$$\begin{cases} I(x,t) = m(t) \circ \rho(p) + n(x,t); \quad & p \in S \\ x = \pi(g(t)p); \quad x \in \pi(g(t)S) \\ I(x,t) = \nu(x,t) \quad & x \mid g^{-1}(t)\pi^{-1}(x) \notin S \end{cases} \tag{30}$$

where $\pi : \mathbb{R}^3 \to \mathbb{R}^2; x \mapsto [x_1/x_3,\ x_2/x_3]^T$ is a central perspective projection. Away from the co-visible portion of the scene S, the image can take any value $\nu(x,t)$. Without loss of generality, the co-visible portion of the scene S can be parametrized as the graph of a function (depth map), $p(x_0) = \bar{x}_0 Z(x_0)$, then the composition of maps

$$w : D \to \mathbb{R}^2;\ x_0 \mapsto x = w(x_0) \doteq \pi(g\bar{x}_0 Z(x_0)) \tag{31}$$

spans the entire group of diffeomorphisms. This is the *motion field*, which is approximated by the optical flow when assumptions (a)-(c) are satisfied. Here a bar $\bar{x} \in \mathbb{P}^2$ denotes the homogeneous (projective) coordinates of the point with Euclidean coordinates $x \in \mathbb{R}^2$. Combining the two equations above, we have the two equivalent representations: $I(w(x_0)) = m \circ \rho(x_0) + n(w(x_0)), \quad x_0 \in w^{-1}(D\backslash\Omega)$, or

$$I(x) = m \circ \rho(w^{-1}(x)) + n(x) \quad x \in D\backslash\Omega \tag{32}$$

with a slight abuse of notation since we have parametrized $\rho : S \to \mathbb{R}^k$ with one of the image planes, via $\rho(x) \leftarrow \rho(p(x))$, and we have re-defined $n(x) \leftarrow n(w^{-1}(x))$. Here D is the domain of the image, and Ω is the subset of the image where the object of interest is not visible (partial occlusion).

It can be shown that m can be eliminated via pre-processing by designing a representation that is a complete invariant statistic, that is a function of the image that is equivalent to it but for the effects of a contrast transformation [10]. There are several such functionals, including the curvature of the level sets of the image, or its dual (the gradient direction), or a normalization of contrast and offset of the image intensity, or spectral ratios if color images are available. In any case, we indicate this pre-processing via

$$\phi(I) = \phi(m \circ I). \tag{33}$$

Correspondence between two image regions can be established when they back-project onto the same portion of the scene S, or when that portion of the scene is *co-visible*. Therefore, establishing correspondence means, essentially, finding a scene (a shape S and an albedo ρ) that, under proper viewing conditions including a motion w and a contrast transformation h, yields a portion of each of the (two or more) images. This can be posed as an optimization problem, which under the assumptions (a)-(b) can be successively reduced into fewer and fewer unknowns:

$$\begin{aligned}
& \arg\min_{m,\rho,g,S} \int_{D\backslash\Omega} |I(x,t) - m \circ \rho \circ \pi(gS)| dx = \ \text{(thm. 7.4, [26])} \\
= \quad & \arg\min_{\rho,w} \int_{D\backslash\Omega} |\phi(I(x,t)) - \phi(\rho \circ w)| dx = \ \text{(thm. 1, [29])} \\
= \quad & \arg\min_{w} \int_{D\backslash\Omega} |\phi(I(x,t)) - \phi(I(x,t-dt) \circ w)| dx
\end{aligned}$$

Of course, the (possibly multiply-connected) region Ω is also unknown, and can be represented via its characteristic function:

$$e_1(x) = \chi(\Omega) \tag{34}$$

where $\chi : D \to \mathbb{R}^+$ is such that $\chi(x) = 1$ if $x \in \Omega$, and $\chi(x) = 0$ elsewhere.

To ease the notational burden, we will assume that contrast has been eliminated via pre-processing, and drop the use of the function ϕ, so we re-define $I \leftarrow \phi(I)$. Writing explicitly the dependency of the "next image" on the occlusion domain, we have

$$\arg\min_{w} \int_{D\backslash\Omega} |I(x,t+dt) - I(w(x,t),t)|^2 dx \tag{35}$$

which is the $\mathbb{L}^2$ component of ψ_{data} in (6).

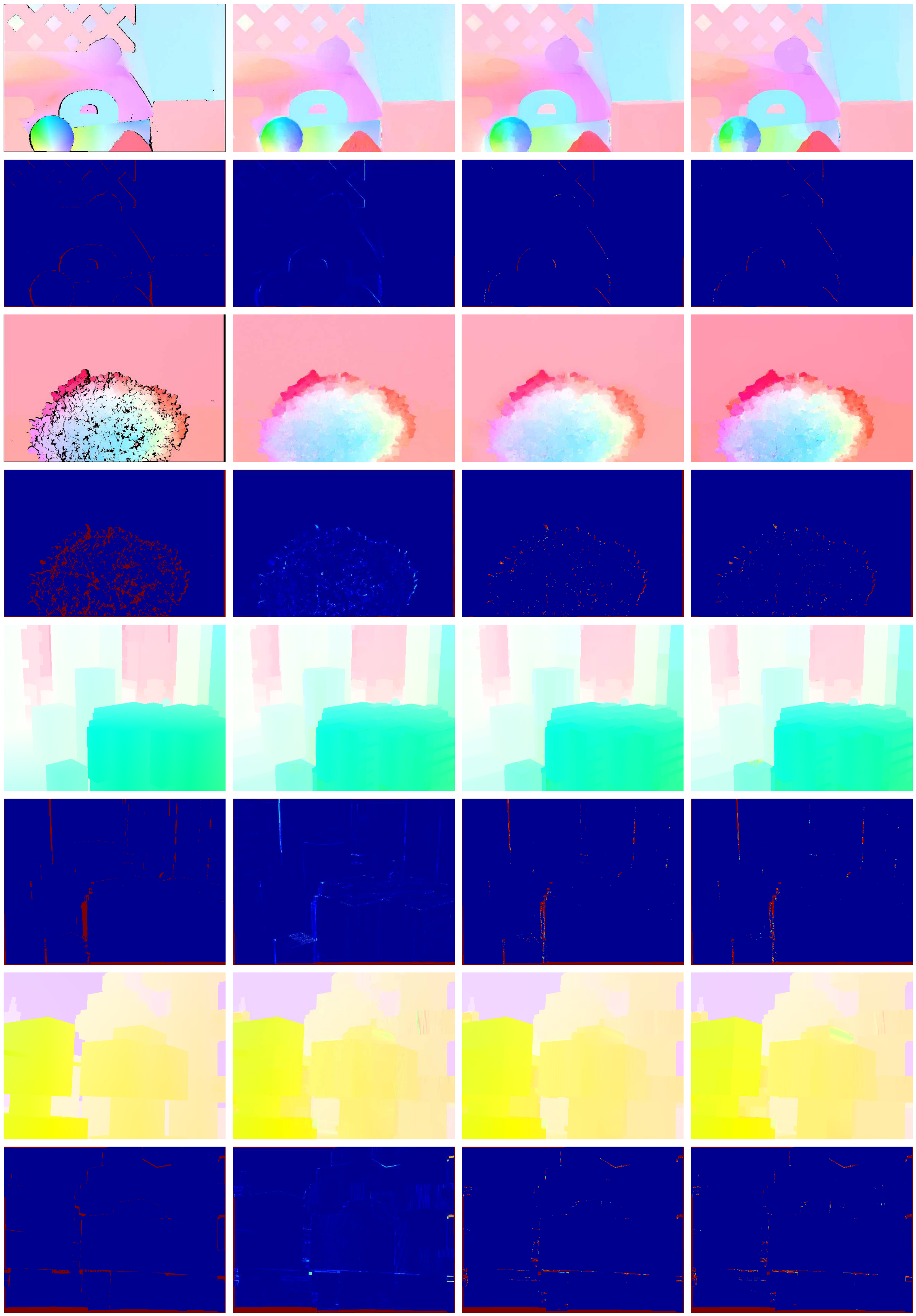


Figure 6: *This figure presents the occlusion and motion estimates on the sequences RubberWhale, Hydrangea, Urban2 and Urban3 from Middlebury dataset. Each sequence occupies two rows. The odd rows, left to right: ground truth optical flow, flow estimates before re-weighting stage and flow estimates after reweighing with Nesterov's algorithm and split-Bregman method. The even rows, left to right: ground truth occluded regions, the initial estimate of the error term* e_1*, the estimate of* $|We_1|$ *after the reweighting step with Nesterov's algorithm and split-Bregman method.*