Abstract

To reliably recover depth, motion, and other physical scene properties from images, low-level vision algorithms need to be able to reason with a local scene model (e.g., a locally planar model for stereo) to address the ambiguity of point-wise estimation. However, since these local models only hold piecewise in a real scene, inference algorithms must also reason about where the model is valid. We introduce a new framework for this reasoning that is defined over a dense set of overlapping regions at multiple scales. Inference in this framework requires finding the right subset of “inlier” regions where the model holds, and the model-based estimates for scene properties in each, through the optimization of an objective that requires these per-region estimates to also be globally consistent. We describe an efficient and parallel architecture to perform this optimization that exploits spatial redundancies in the overlapping set of regions. Specifically, we show that when these regions are organized into a multi-scale hierarchy, inference only requires sharing information across a very sparse set of links between parent and child regions. We demonstrate the benefits of this framework by applying it to stereo estimation, and evaluating performance on the KITTI benchmark.

1. Introduction

Low-level vision is the estimation of depth, motion, shape, and other physical scene properties from visual measurements. Since it is ill-posed, methods often employ a local model that is expected to apply piecewise across the scene, and that restricts the variation of scene values within a piece. Slanted planes for binocular disparity, affine optical flows, and families of smooth shapes for surface normals are common examples. The restriction on scene variability in applicable regions allows image cues to be aggregated spatially, thereby reducing the ambiguity that exists point-wise. The fundamental challenge lies in identifying—automatically from the image input—the sizes and shapes of the aggregation regions that are right for each part of a scene. Regions that are too small do not sufficiently reduce the underlying ambiguity, while those that are too big or the wrong shape span abrupt scene changes that violate the local model and make estimates unreliable (e.g., Fig. 1 (a)).

We introduce a new inference framework that addresses this challenge. It explicitly considers a large set of dense, overlapping regions of many sizes that redundantly cover the image plane (Fig. 1 (b)). It then attempts to determine which regions are inliers to the local model, and the model-based estimates of their scene values. This is done based on an objective that requires each inlying region to be consistent with: 1) the relevant portion of the input image data; and 2) the scene estimates of all other inlying regions that overlap with it. The output of the framework—the inlier statuses of all regions and the local estimates from the inliers—offers a rich representation of the physical scene. In particular, it can be used to read off a consistent global scene map, formed through the consensus of inlying regions, as well as a point-wise measure of confidence, given by the number of inliers that include each point.

In comparison to traditional approaches based on Markov random fields (MRFs), the consensus framework reasons in a much larger variable space, and more critically, with orders of magnitude more links between variables. This is because it enforces simultaneous consistency between thousands of regions that overlap any single pixel. Despite this apparent greater complexity, two properties of the framework make inference not only feasible, but efficient. First, since the dense region-set embodies an overcomplete scene representation, i.e. with many more internal variables than there are values in the output scene map, good solutions can be reached by a simple alternating algorithm that is similar to expectation-maximization. Second, we show analytically that when the regions are organized hierarchically by scale (e.g. Fig. 1 (c)), each region only needs to sum information from its parents and children (Fig. 1 (d)). This leads to a significant reduction in the amount of required computation, since these hierarchical links constitute only a minuscule fraction of the total consistency-based links induced by the consensus objective.

The inference architecture ends up being composed of a large network of computational units, one for each re-
Figure 1. The Consensus Framework. (a) Inference in low-level vision with local models for physical scene properties (such as slanted planes for disparity) allow aggregation of underlying image cues to make estimation reliable. But they also require determining the right shapes and sizes of these aggregation regions in different parts of the scene, with the goal being to select the largest valid regions where the model holds. (b) We introduce an inference framework that reasons with a dense set of overlapping regions at multiple scales, to select the right valid subset of these regions and generate estimates from each that are globally consistent. (c) We visualize the links induced by the global consistency objective in one dimension, with a dense region set that contains all overlapping patches of size \(2^k\), for multiple scales \(k\). Our framework requires agreement between the scene estimates at every point from all valid, or inlying, regions that include it to agree. This objective links each point in the image plane to the large set of regions that include it (shown in blue), where each region is a member of several such sets (red). To make inference efficient, we exploit the spatial redundancy in the framework by organizing its regions in a multi-scale hierarchy. Each region in this hierarchy is linked to a small number (two, in this example) of non-overlapping child regions, from a smaller scale, that spatially partition the parent. (d) This leads to a distributed inference architecture where information need only be shared across the much sparser set of hierarchical links, with computational units—one for each region—iteratively performing the same set of computations, and collaborating to produce a consistent scene representation.

2. Related Work

Many techniques have been developed binocular stereo, optical flow, shape-from-shading, and other low-level vision problems. These each produce a dense scene map that contains a scene value (depth, motion, surface normal, etc.) at each location in the image plane. While these techniques vary greatly in the way they derive information point-wise from image cues, the mechanisms they use for spatial aggregation typically follow one of three different paradigms.

One paradigm is local. It uses intensity and texture information to explicitly define one spatial support region around each pixel. One flavor of these methods (e.g., [9]) makes an independent determination of the support region for each pixel, without requiring overlapping regions from neighboring locations to be consistent. Cost-volume filtering methods for stereo and optical flow also fall into this category [14, 28]. Other methods (e.g., [18, 23, 25]) define these supports as a non-overlapping partition of spatial locations, usually computed through a segmentation algorithm (at some chosen scale) that accounts for consistency in intensities as well as scene map values within a region.

Variational methods form another category in which methods find the scene map that minimizes a per-pixel data cost along with spatial regularization in the form of...
a penalty on the spatial derivatives of the scene map \([3, 4, 8, 10, 13, 15]\). With an appropriately chosen combination of derivative filters, this regularization strategy can be thought of as penalizing deviations from a local model (locally constant, affine, etc.). While the penalty is applied to derivatives at all points, it is usually chosen in the form of an \(L_1\) norm to promote piecewise adherence to the local model, and to avoid smoothing across true discontinuities. Some of these methods also use multi-scale reasoning to promote convergence to better local optima of the objective and to speed up inference. This is done either by employing a sequential coarse-to-fine approach, where solutions from a coarser scale are used to initialize the minimization at the next finer scale (e.g. [4]), or by simultaneously penalizing derivatives at multiple scales [1].

The third dominant paradigm are MRF techniques. Unlike variational methods that try to implicitly guide the scene map to have few but sharp discontinuities, MRF-based methods [11, 17, 20, 22, 23, 24] do so explicitly. Cliques in the MRF, consisting of a small number of adjoining nodes, are defined to make hard decisions on whether the local model applies across them (i.e., whether they belong to the same “piece” of the local model). However, since these models involve continuous label spaces and non-submodular smoothness terms, solving them requires computationally expensive approximate inference algorithms. Some state-of-the-art approaches [22, 23, 24] work with MRFs defined over super-pixels rather than individual pixels, since reasoning with the smaller number of super-pixels increases the likelihood of being able to find a good solution within a reasonable execution time. However, the choice of the number of super-pixels also imposes a hard selection of scale for the scene—since the local model is now constrained to hold within each super-pixel, structures too small to be assigned their own super-pixel are smoothed over.

The consensus framework we introduce is different from traditional MRF approaches in that it is formulated over overlapping regions at multiple scales and involves checking consistency between all set of overlapping regions. In this respect, it is similar to the formulation of the shape-from-shading method recently introduced in [21]. In this paper, we generalize the approach to define an objective function that applies to a variety of low-level vision processes and local models, and show that this objective can be optimized very efficiently through an appropriate encoding of the model and by leveraging a multi-scale hierarchical architecture.

3. Framework Elements

We begin with a formal description of the three main components of the proposed framework. First, there is the global scene map. This is a function \(Z(n) \in \mathbb{R}^d\) on the two-dimensional image plane, with \(n = (x, y)\) indexing discrete spatial locations. \(Z(n)\) may be scalar-valued (\(d = 1\)) for properties such as stereo disparity, or vector-valued for properties such as motion and 3D surface orientation.

Second, there is a dense set \(P\) of overlapping regions \(p \in P\) within the image plane, each one a collection of locations \(n\). Set \(P\) has regions at \(K\) different scales, and symbol \(P_k\) represents the subset of regions at scale \(k \in \{1 \ldots K\}\). By convention, larger values of \(k\) corresponds to larger regions, and the regions are organized hierarchically: Any region \(p \in P_k\) at scale \(k > 1\) is the union of a subset, denoted \(H_p\), of non-overlapping child regions from scales smaller than \(k\). Figure 1 (c,d) shows an example of such a region set for a one-dimensional image plane, where each \(P_k\) is the set of overlapping regions of length \(2^k\), and each \(p \in P_k, k > 1\) is the union of two children from \(P_{k-1}\).

The final component is the local model. It is expected to apply piecewise across most of the scene, and it restricts accordingly the allowable choices for scene values within any region \(p\). It is encoded in a mathematical form that encompasses all sorts of local models proposed in computer vision [2, 21], while also enabling the system to exploit the computational redundancy inherent to a hierarchy of regions. The local model says that scene values within any region \(p\) must satisfy:

\[
Z(n) = U(n)\theta_p, \quad \forall n \in p, \tag{1}
\]

where \(U(n) \in \mathbb{R}^{d \times M}\) is some pre-defined matrix-valued function on the image plane, and \(\theta_p \in \mathbb{R}^M\) is a variable associated with region \(p\). Algebraically, this restricts local scene values to an \(M\)-dimensional linear sub-space, regardless of region size; and as a consequence of using a common \(U(n)\), local scene estimates from two overlapping regions \(p\) and \(p'\) agree whenever \(\theta_p = \theta_{p'}\). Here are some examples of functions \(U(n)\) and their corresponding physical interpretations:

\[
U(n) = \begin{bmatrix} x & y & 1 \end{bmatrix}, d = 1, M = 3,
\]

(disparity of locally-planar surfaces),

\[
U(n) = \begin{bmatrix} \partial/\partial x \\ \partial/\partial y \end{bmatrix} \begin{bmatrix} x^2 & y^2 & xy & x & y \end{bmatrix}, d = 2, M = 5,
\]

(normals of locally-quadratic surfaces),

\[
U(n) = \begin{bmatrix} x & y & 1 \\ 0 & x & y & 1 \end{bmatrix}, d = 2, M = 6,
\]

(flow vectors for locally-affine motion).

With the three components in hand, inference requires determining: a) which regions \(p \in P\) are inliers with respect to the local model; and b) for all inlying regions, values of the per-region variables \(\theta_p\) that are supported by the image data and consistent with each other. Inliers are indicated by a binary variable \(I_p \in \{0, 1\}\) associated with each patch. Once inferred, the values of \(\{I_p, \theta_p\}\) together

provide a rich and over-complete representation of the physical scene. At each point \( n \), local grouping information is available through the subset \( J_n \) of (potentially thousands of) inlying regions covering that point:

\[
J_n = \{ p \mid p \ni n, I_p = 1 \}.
\]

An estimate \( \tilde{Z} \) of the global scene map is induced as the point-wise average, or consensus, of the local estimates from inlying regions:

\[
\tilde{Z}(n) = \frac{1}{|J_n|} \sum_{p \in J_n} U(n)\theta_p = \frac{1}{\sum_{p \ni n} I_p} \sum_{p \ni n} U(n)\theta_p I_p. \tag{3}
\]

The count \( |J(n)| \) represents the degree of consensus at each point, and provides a point-wise measure of confidence in the estimate \( \tilde{Z} \).

Inference is then cast as a minimization of the following cost over variables \( \{I_p, \theta_p\} \):

\[
L(\{I_p, \theta_p\}_{p \in P}) = \sum_{p : I_p=0} \tau_p + \sum_{p : I_p=1} \lambda \sum_{n \in p} \sum_{p \ni n} \sum_{I_p \ni n} \sum_{\theta_p} \left[ \sum_{\theta_p} D_p(\theta_p) \right] + \lambda \sum_n |J_n| \text{Var} \left[ \{U(n)\theta_p\}_{p \in J_n} \right]. \tag{4}
\]

The first term applies a cost \( \tau_p \) for declaring region \( p \) an outlier, in line with intuition that the local model is often valid. The second term scores local variables \( \theta_p \) in each inlying region using data cost \( D_p(\cdot) \), typically measuring the ability of restricted local scene estimates \( U(n)\theta_p, \forall n \in p \) to explain the relevant image data. Both \( \tau_p \) and \( D_p(\cdot) \) can optionally be augmented to encode prior information about the scene or context from semantic visual processes. The final \( \lambda \)-weighted term promotes consistency between overlapping regions by penalizing, at every point, the variance of the scene predictions from inlying regions that cover it.

### 4. Inference Algorithm

To minimize (4), we re-write the consistency term in terms of the global scene map \( Z \), creating a related cost \( L' \):

\[
L'(\{I_p, \theta_p\}_{p \in P}, Z) = \sum_{p : I_p=0} \tau_p + \sum_{p : I_p=1} \left( D_p(\theta_p) + \lambda \sum_n \|U(n)\theta_p - Z(n)\|^2 \right), \tag{5}
\]

where the two costs are equal when \( Z \) is set to the consensus, i.e. \( L'(\{I_p, \theta_p\}, Z) = L(\{I_p, \theta_p\}) \). In this new cost, the per-region summations are quadratic functions of variables \( \theta_p \):

\[
C_p(\theta, Z) = \sum_{n \in p} \|U(n)\theta_p - Z(n)\|^2 = \theta_p^T Q_p \theta_p - 2\phi_p^T \theta_p + \epsilon_p, \tag{6}
\]

with each \( Q_p = \sum_{n \in p} U(n)^T U(n) \) a pre-computed \( M \times M \) matrix permanently associated with region \( p \) and each \( \phi_p, \epsilon_p \) an \( M \)-vector and a scalar, respectively, derived from \( Z \) as:

\[
\phi_p = \sum_{n \in p} U(n)^T Z(n), \quad \epsilon_p = \sum_{n \in p} ||Z(n)||^2. \tag{7}
\]

Cost \( L' \) is minimized iteratively, with each iteration having two steps. The first step is a minimization over region variables \( \{I_p, \theta_p\} \) with \( Z \) fixed. Conveniently, this can be done independently—and in parallel—for each region since there are no cross-region terms in \( L' \) when \( Z \) is fixed. These independent minimizations are achieved by setting

\[
\theta_p = \arg \min_{\theta_p} [D_p(\theta_p) + \lambda C_p(\theta, Z)], \tag{8}
\]

and then,

\[
I_p = \begin{cases} 
0, & \text{if } [D_p(\theta_p) + \lambda C_p(\theta_p, Z)] > \tau_p, \\
1, & \text{otherwise .} 
\end{cases} \tag{9}
\]

In other words, the best model-based explanation is found for each region \( p \), and then the region is declared outlier if the error-of-fit exceeds the outlier cost \( \tau_p \).

The second step at each iteration is a minimization over \( Z \) with region variables fixed at their new values. This is achieved simply by setting \( Z = \tilde{Z} \) as per (3), and it is thus guaranteed that \( L(\{I_p, \theta_p\}) = L'(\{I_p, \theta_p\}, Z) \) at the end of each iteration. Consequently, beginning with any initial estimate of the scene map \( Z \), each iteration decreases the value of \( L' \), and therefore of \( L \), which converges to a (local) minimum whenever \( \{D_p(\cdot)\} \) have finite lower bounds.

Convergence to a good local minimum is promoted by beginning the iterations with a smaller value for the consistency weight \( \lambda \), and then increasing it to its final value across the initial iterations. Interestingly, this induces a temporal coarse-to-fine refinement of the scene map during inference. Early-on, smaller \( \lambda \) values allow more inlying regions, causing the consensus to be smoothed across larger areas. As \( \lambda \) increases, more regions that span scene discontinuities become outliers, and the consensus exhibits progressively finer detail.

### 4.1. Hierarchical Computation

The computational cost of inference depends on the complexities of the three parts of every iteration:

1. **Computing intermediate regional consistency terms \( \{\phi_p, \epsilon_p\} \) from \( Z \) as per (7).**
2. **Updating \( \{\theta_p, I_p\} \) for every region \( p \) as per (8),(9).**
3. **Setting \( Z = \tilde{Z} \) as per (3).**

The complexity of the second part depends on the forms of functions \( D_p(\cdot) \), but it usually scales well because it involves parallel optimization over \( M \)-dimensional domains (regardless of region size). The first and third parts would
scale poorly if implemented naively, but this can be averted by exploiting the hierarchical structure of \( P \).

First, consider the computation of \( \{ \phi_p, e_p \} \) from \( Z \). Using the fact that every region \( p \) is partitioned by its child regions \( c \in H_p \), we can write

\[
\phi_p = \sum_{n \in p} U(n)^T Z(n) = \sum_{c \in H_p} \sum_{n \in c} U(n)^T Z(n) = \sum_{c \in H_p} \phi_c,
\]

and similarly, \( e_p = \sum_{c \in H_p} e_c \). This reduces the number of additions significantly—from the size of the region \( p \) to just the number of its children. To ensure that values of \( \{ \phi_c, e_c \} \in H_p \) are available, calculations of \( \phi_p, e_p \) are scheduled in an upward sweep through the hierarchy, using explicit summation over \( n \) for regions scale \( k = 1 \), and the cheaper right-most expression of (10) for progressively larger scales.

The hierarchical structure can also be leveraged to efficiently compute the consensus \( Z \) from the current values of the region variables \( \{ \theta_p, I_p \} \). Let \( H_p^{-1} = \{ r : H_r \supseteq p \} \) denote the set of parents for any region \( p \). We then recursively define augmented variables \( \{ \theta_p^+, I_p^+ \} \) for every region \( p \) as:

\[
\theta_p^+ = \theta_p + \sum_{r \in H_p^{-1}} \theta_r^+, \quad I_p^+ = I_p + \sum_{r \in H_p^{-1}} I_r^+,
\]

which can be through a downward sweep across the pyramid. Then, it is easy to show that the numerator and denominator of the expression for \( Z(n) \) in (3) are given by:

\[
\sum_{p \ni n} U(n) \theta_p I_p = U(n) \sum_{\{ p \ni n \} \cap |I_p|} \theta_p^+,
\]

\[
\sum_{p \ni n} I_p = \sum_{\{ p \ni n \} \cap |I_p|} I_p^+.
\]

Therefore, instead of computing summations over all overlapping regions at all scales for each \( n \), the consensus can be computed using summations over the augmented variables \( \{ \theta_p^+, I_p^+ \} \) of regions at just the smallest scale.

The gains from using these recursive computations is substantial, and can be interpreted as reducing the effective connectivity of the framework to just the sparse set of hierarchical links. For the network in Fig. 1 (c,d), it represents a reduction, in the number of required summations for (3) and (7), from \( O(2^K N) \) to \( O(KN) \). Moreover, while the recursion requires different scales to be processed sequentially, note that the computations in (10) and (11) can still be carried out for all regions \( p \in P_k \) at each scale \( k \) in parallel.

Therefore, as visualized in Fig. 1 (d), inference happens in a distributed architecture, requires the identical computations of (8), (9), (10), and (11) at each region, with computations at each scale happening in parallel and information being passed through hierarchical links between scales—all of which arises naturally as an efficient way to optimize a well-defined mathematical objective. A complete listing of the algorithm is included in Appendix A.

5. Application to Stereo

In this section, we describe the application of the consensus framework to the task of stereo estimation, and its evaluation on the KITTI benchmark [6]. We reason with a planar local model (i.e., \( U(n) = [x \ y \ 1] \)), and define our region set \( P \) to be a two-dimensional equivalent of the pyramid in Fig. 1(b,c), with five scales consisting of all overlapping patches of sizes \( 4 \times 4, 8 \times 8, 16 \times 16, 32 \times 32 \), and \( 64 \times 64 \), where patches at all but the finest scale can be partitioned into four children from the next lower scale.

We follow the approach of [22, 23, 24] in defining our data cost \( D_p(\cdot) \). In particular, we use the implementation from Yamaguchi et al. [24], that implements semi-global matching (SGM) [7] with a cost based on absolute differences of gradients and the Census transform [26], to compute an initial set of approximate disparity estimates \( Z_{\text{som}}(n) \) at a semi-dense set of locations \( n \in \Omega_{\text{SGM}} \). The data costs for every region \( p \) are then defined as:

\[
D_p(\theta) = \sum_{n \in p} w_{\text{som}}(n) \left( U(n) \theta - Z_{\text{som}}(n) \right)^2,
\]

where \( w_{\text{som}}(n) = 0 \) if \( n \notin \Omega_{\text{som}} \), \( 1/4 \) if there is a discontinuity in \( Z_{\text{som}} \) around \( n \), and 1 otherwise. Note that since each \( D_p(\cdot) \) above is also a quadratic function of \( \theta \), the minimizations in (8) simply involve solving a \( 3 \times 3 \) linear system for each region (and moreover, we can re-use the results of LDL decompositions across iterations when \( \lambda \) is constant).

The scalar outlier costs \( \{ \tau_p \} \) for various patches are defined to be proportional to their size, and to a factor that is based on intensity variance within each patch:

\[
\tau_p = \tau_0 \times |p| \times \max \left( 0.5, \exp \left( -0.25 V_p^2 \right) \right),
\]

where \( V_p \) is a count of the number of patches that share a child patch with \( p \) (there are eight such patches), and whose intensity variance is lower than that of \( p \), and we set \( \tau_0 = 1.44 \). For the consistency weight, we use \( \lambda = 0.4 \), and as described in Sec. 4, we begin the iterations with a lower value of \( \lambda' = 2^{-18} \lambda \), and increase it by a factor of eight every six iterations till it reaches its final value.

Finally, we add a way to incorporate reasoning about occlusions. While one could achieve this encoding using more sophisticated definitions of \( D_p(\cdot) \) or \( \tau_p \), we find that a much simpler approach suffices. We use the fact that the pixels missing in \( \Omega_{\text{som}} \) correspond to those that have failed a left-right consistency check, and are likely occluded. Using the data costs defined as above, the consensus framework usually yields a scene map where disparity values in occluded

\[1\] Note that for a region \( p \in P_k \) at the largest scale, \( H_p^{-1} = \emptyset \).
regions, in the absence of any input data, are interpolated between the occluded and occluding planes either smoothly, or with a discontinuity at an arbitrary location within the region. In order to incorporate the intuition that occluded pixels are likely to be part of the occluded background, we run the alternating minimization for fifty iterations; then set the values of the consensus $Z(n)$, at the potentially occluded points in $n \notin \Omega_{\text{SGM}}$, to the lower of their current value and that of the background pixel on the same epipolar line (i.e., the nearest pixel on the same horizontal line in $\Omega_{\text{SGM}}$); and then run the minimization for another thirty iterations.

A reference implementation of this stereo algorithm, designed to make use of thread-based parallelism and SIMD instruction sets, takes an average of only six seconds (1.5 seconds for the initial SGM step) on a $1240 \times 370$ image, when running on a CPU with six cores. Moreover, since the computations in the framework have a degree of parallelism roughly equal to the resolution of the input image (computations for all regions at a scale can always be done in parallel), we expect execution time will continue to decrease with access to larger numbers of cores.

5.1. Evaluation on KITTI

We evaluate the proposed algorithm on the KITTI benchmark [6] which contains a total of 389 grayscale image pairs of rural road scenes, captured using an autonomous driving
platform equipped with a pair of high-resolution cameras. A Velodyne laser scanner provides ground truth at a subset of pixels in each scene. This ground truth is made available for a subset of 194 image pairs—the training set—and withheld for the remaining image pairs that form the testing set. A website associated with the database tracks the performance of stereo algorithms on the testing set. Note that while the benchmark also contains temporally-adjacent stereo frames that allow simultaneous reasoning about optical flow and stereo, we ignore those extra frames and consider the pure stereo problem here.

Figure 2 visualizes various aspects of the internal representation of our framework on convergence, for three scenes in the KITTI training set. The top row shows the consensus global disparity map, and Rows 2–6 visualizes a regularly-spaced subset of in the inlier statuses $I_p$. Row 7 provides another view of variables $I_p$, by explicitly showing some of the “support regions” formed as the union of all patches in $J_n$, for various pixels $n$. These regions by-and-large group together points whose disparity values would be well-explained by a slanted plane model. As expected, there is significant variation in the size and shapes of the support regions across each scene, matching the scale of the underlying scene structures. This highlights the distinction from superpixel-based MRF approaches [22, 23, 24], which require choosing a single scale for the entire scene. Also note that for many pairs of points that do not directly lie in each others’ support regions, the regions themselves have significant overlap. Through such overlap, the consensus estimate at a point has benefited from aggregation across regions that are larger than the union of the set of patches that include it.

The final row in Fig. 2 visualizes the degree of consensus $|J_n|$ at all points (blue saturation), simultaneously with locations of erroneous estimates (red). We see that many of the errors occur around object boundaries and near small scene structures, which are also points where $|J_n|$ is low. We quantify this observation in Fig. 3, and find that average estimation error drops rapidly as we discard points with the lowest values of $|J_n|$. This an other benefit of the rich internal representation: In addition to providing a global scene estimate, it also provides a natural measure of point-wise confidence in this estimate.

Next, we visualize the progression of the alternating minimization algorithm. Figure 4 shows the evolution of the consensus $Z(n)$ across iterations. In the early iterations as the consistency weight $\lambda$ is increased toward its final value, the map undergoes a coarse-to-fine refinement, with image boundaries becoming sharper. After the fiftieth iteration, the occlusion-based correction is applied to propagate disparity estimates from background planes into potentially-occluded regions. Since this is a simple correction applied independently along each epipolar line, it introduces inconsistencies within occluded regions. The scene map therefore benefits from further refinement through another thirty iterations of minimizing the consensus cost, yielding a final estimate that is more accurate.

Table 1 compares the consensus framework with other state-of-the-art stereo algorithms in terms of various error quantiles on the KITTI testing set. The most direct comparisons of our results are with those of [19, 22, 23, 24], since these methods all use the same approach to derive their data costs (census transform and gradient-based matching with SGM). These only differ—from us, and from each other—in their approach to spatial aggregation. The consensus framework outperforms all of these methods on all error metrics, while also having a low execution time.

Table 1 also reports the performance of the recently released MC-CNN [27] algorithm, which computes point-wise matching costs using a multi-layer convolutional neural network. This produces lower error values than all other methods, including ours, in exchange for greater computation (the method takes 100 seconds on a GPU with 2880 CUDA cores). This is encouraging, because improved pixel-wise data costs like this one can be directly substituted into the consensus framework to enhance accuracy.

Finally, we demonstrate the benefit of the pixel-wise confidence measure in our framework by reporting a second set of results in Table 1. This is simply produced by discarding a small number of pixels with the lowest degree of consensus $|J_n|$ (i.e. those with degree less than 200 out of the maximum possible value of $\approx 5500$). This second set of error quantiles—computed now on the high-confidence set with only $\approx 3.6\%$ fewer pixels—are the smallest of all methods. In this mode, the proposed algorithm efficiently produces very reliable disparity estimates, at all but a small fraction of locations. This also suggests a strategy for leveraging sophisticated matching strategies such as [27] when

---

2This table includes only pure stereo methods, those that use just one stereo pair as input.
Figure 4. Evolution of the scene map across iterations. The first two frames correspond to iterations when the consistency weight $\lambda$ is still being increased towards its final value, causing the map to undergo a coarse-to-fine refinement. After iteration #50, we apply the occlusion-based correction step, and then run the minimization for another 30 iterations.

| Method       | Avg. Error | $>2\text{px}$ | $>3\text{px}$ | $>4\text{px}$ | $>5\text{px}$ | Exec. Time |
|--------------|------------|----------------|----------------|----------------|----------------|------------|
|              | All       | NOC            | All            | NOC            | All            | NOC        |            |
| ATGV [13]    | 1.6px     | 1.0px          | 9.05%          | 7.08%          | 6.88%          | 5.02%      | 5.76%      | 3.99%      | 5.01%      | 3.33%      | 6min: 8 cores |
| wSGM [16]    | 1.6px     | 1.3px          | 8.72%          | 7.27%          | 6.18%          | 4.97%      | 4.89%      | 3.88%      | 4.11%      | 3.25%      | 6: 1 core    |
| AARBM [5]    | 1.2px     | 1.0px          | 8.70%          | 7.36%          | 5.94%          | 4.86%      | 4.56%      | 3.67%      | 3.69%      | 2.96%      | 0.25s: 1 core |
| *PCBP [22]   | 1.1px     | 0.9px          | 7.62%          | 5.08%          | 5.37%          | 4.04%      | 4.29%      | 3.14%      | 3.64%      | 2.64%      | 5min: 4 cores |
| *StereoSLIC [23] | 1.0px     | 0.9px          | 7.20%          | 5.76%          | 5.11%          | 3.92%      | 4.04%      | 3.04%      | 3.33%      | 2.49%      | 2.3s: 1 core |
| *DDS-SS [19] | 1.0px     | 0.9px          | 6.96%          | 5.91%          | 4.59%          | 3.83%      | 3.49%      | 2.90%      | 2.83%      | 2.36%      | 1min: 1 core |
| *PCP-B-S [23] | 1.0px    | 0.8px          | 6.75%          | 5.19%          | 4.72%          | 3.40%      | 3.75%      | 2.62%      | 3.15%      | 2.18%      | 5min: 1 core |
| *SPS-St [24] | 1.0px     | 0.9px          | 6.28%          | 4.98%          | 4.41%          | 3.39%      | 3.52%      | 2.72%      | 3.00%      | 2.33%      | 2s: 1 core   |
| MC-CNN [27]  | 1.0px     | 0.8px          | 5.39%          | 4.30%          | 3.84%          | 2.61%      | 3.01%      | 2.04%      | 2.52%      | 1.75%      | 100s: GPU    |
| *Proposed: All n | 0.9px | 0.8px          | 5.88%          | 4.85%          | 4.10%          | 3.30%      | 3.26%      | 2.59%      | 2.74%      | 2.16%      | 6s: 6 cores  |
| Only $|J_n| \geq 200$ (96.4% density) | 0.8px | 0.6px          | 4.59%          | 3.50%          | 2.98%        | 2.14%      | 2.26%      | 1.56%      | 1.85%      | 1.24%      |

*: Same Matching Cost

Table 1. Comparison with the state-of-the-art on the KITTI testing set. Performance is measured in terms of average error, as well as percentage of estimates with error greater than different thresholds. For each metric, the “All” column reports values computed over all ground truth pixels, and “NOC” over only those those that are within the field-of-view of the right camera. The last row reports the accuracy of our method’s estimates that have confidence measure $|J_n|$ above a threshold, and correspond to errors values computed over 96.4% of the points with ground truth available.

execution time is a bottleneck (such as for automated driving applications)—one where the more expensive matching costs are computed only for the small number of low-confidence pixels.

6. Conclusion

In this paper, we introduced a framework for low-level visual inference with local scene models that reasons with a large overlapping, multi-scale set of regions, to determine which of them are outliers, and which of them can generate model-based scene value estimates while being consistent with each other. Despite the larger variable space, and the greater complexity of the consensus objective, we showed that inference can be achieved efficiently by recognizing that the regions can be organized hierarchically. An evaluation on stereo estimation found that the framework outperforms existing approaches to spatial reasoning.

An important direction of future research lies in applying the framework to problems involving estimating different physical properties of the same scene (such as material and shape), with different piecewise local models for each, when the aggregation regions of one property suggest, but do not determine, those of the other.

On a different note, many properties of the consensus framework—multi-scale collaboration, implementation as a distributed architecture of computational units carrying out the same operations, coarse-to-fine evolution of the scene map, etc.—mimic behavior observed in biological systems [12]. It would be interesting to explore these links systematically—to investigate whether the framework, or some variation of it, can serve as a faithful model for biological processing; as well as whether insights from biology can be used to further improve the framework.

References

[1] J. T. Barron and J. Malik. Shape, illumination, and reflectance from shading. Technical Report UCB/EECS-2013-117, EECS, UC Berkeley, May 2013.
[2] S. Birchfield and C. Tomasi. Multiway cut for stereo and motion with slanted surfaces. In Proc. CVPR, 1999. 3
[3] K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. SIAM J. on Imag. Sc., 3(3):492–526, 2010. 3
[4] A. Bruhn, J. Weickert, T. Kohlberger, and C. Schnörr. A multigrid platform for real-time motion computation with discontinuity-preserving variational methods. IJCV, 2006. 3
[5] N. Einecke and J. Eggert. Block-matching stereo with relaxed fronto-parallel assumption. In Proc. IEEE Intelligent Vehicles Symposium, 2014. 8
[6] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In Proc. CVPR, 2012. 2, 5, 6
[7] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. IEEE Trans. PAMI, 2008. 5
[8] B. K. Horn and M. J. Brooks. The variational approach to shape from shading. Comp. Vision, Graph., and Imag. Proc., 1986. 3
[9] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: Theory and experiment. IEEE Trans. PAMI, 1994. 2
[10] G. Kuschk and D. Cremers. Fast and accurate large-scale stereo reconstruction using variational methods. In Proc. ICCV Workshops, 2013. 3
[11] V. Lempitsky, S. Roth, and C. Rother. Fusionflow: Discrete-continuous optimization for optical flow estimation. In Proc. CVPR, 2008. 3
[12] M. D. Menz and R. D. Freeman. Functional connectivity of disparity-tuned neurons in the visual cortex. J. of Neurophysiology, 2004. 8
[13] R. Ranftl, T. Pock, and H. Bischof. Minimizing tgv-based variational models with non-convex data terms. In Scale Space and Variational Methods in Comp. Vision. 2013. 3, 8
[14] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. In Proc. CVPR, 2011. 2
[15] N. Slesareva, A. Bruhn, and J. Weickert. Optic flow goes stereo: A variational method for estimating discontinuity-preserving dense disparity maps. In Pattern Recog. 2005. 3
[16] R. Spangenberg, T. Langner, and R. Rojas. Weighted semi-global matching and center-symmetric census transform for robust driver assistance. In Computer Analysis of Images and Patterns, 2013. 8
[17] J. Sun, N.-N. Zheng, and H.-Y. Shum. Stereo matching using belief propagation. IEEE Trans. PAMI, 2003. 3
[18] C. Vogel, S. Roth, and K. Schindler. Piecewise rigid scene flow. In Proc. ICCV, 2013. 2
[19] D. Wei, C. Liu, and W. Freeman. A data-driven regularization model for stereo and flow. In Proc. Intl. Conf. on 3D Vision, 2014. 7, 8
[20] O. Woodford, P. Torr, I. Reid, and A. Fitzgibbon. Global stereo reconstruction under second-order smoothness priors. IEEE Trans. PAMI, 2009. 3
[21] Y. Xiong, A. Chakrabarti, R. Basri, S. J. Gortler, D. W. Jacobs, and T. Zickler. From shading to local shape. IEEE Trans. PAMI, 2014. 3
[22] K. Yamaguchi, T. Hazan, D. McAllester, and R. Urtasun. Continuous markov random fields for robust stereo estimation. In Proc. ECCV. 2012. 3, 5, 7, 8
[23] K. Yamaguchi, D. McAllester, and R. Urtasun. Robust monocular epipolar flow estimation. In Proc. CVPR, 2013. 2, 3, 5, 7, 8
[24] K. Yamaguchi, D. McAllester, and R. Urtasun. Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In Proc. ECCV, 2014. 3, 5, 7, 8
[25] K.-J. Yoon and I. S. Kweon. Adaptive support-weight approach for correspondence search. IEEE Trans. PAMI, 2006. 2
[26] R. Zabih and J. Woodfill. Non-parametric local transforms for computing visual correspondence. In Proc. ECCV, 1994. 5
[27] J. Žbontar and Y. LeCun. Computing the stereo matching cost with a convolutional neural network. arXiv:1409.4326 [cs.CV], 2014. 7, 8
[28] K. Zhang, Y. Fang, D. Min, L. Sun, S. Y. Yan, Q. Tian, et al. Cross-scale cost aggregation for stereo matching. In Proc. CVPR, 2014. 2
Appendix A: Inference Algorithm Listing

We present here a summary of the inference algorithm for reference. It takes as input the following elements:

1. Sets of regions \( P_k, k \in \{1, \ldots, K\} \) at \( K \) different scales.
2. For each region \( p \in P_k, k > 1 \), a set \( H_p \) consisting of non-overlapping child regions that partition \( p \), and are from scales smaller than \( k \) (i.e. \( H_p \subset \bigcup_{k'=1}^{k-1} P_{k'} \)).
3. The data cost functions \( D_p(\cdot) \) and scalar outlier costs \( \tau_p \) for every region \( p \).
4. The value of the consistency weight \( \lambda \). Additionally, its value \( \lambda_0 \) to be used at the beginning of the iterations, the factor \( \lambda_f > 1 \) by which it is to be increased at every \( T_\lambda \) iterations.
5. An initial estimate \( Z_0(n) \) of the scene value map.

Given these elements, the algorithm to minimize the cost function \( L \) is reproduced below:

```plaintext
# Initialization
Set \( \lambda^* = \lambda_0, Z(n) = Z_0(n) \) for all \( n \).
In parallel, for all \( p \in P_1 \):
  Set \( Q_p = \sum_{n \in p} U(n)^T U(n) \).
end;
for \( k = 2 \ldots K \)
  In parallel, for all \( p \in P_k \):
    Set \( Q_p = \sum_{c \in H_p} Q_c \).
end;
# Main Iterations
for iters = 1 \ldots \text{MAXITERS}
  # Upsweep
  In parallel, for all \( p \in P_1 \):
    Set \( \phi_p = \sum_{n \in p} U(n)^T Z(n), e_p = \sum_{n \in p} \|Z(n)\|^2 \).
  end;
  for \( k = 2 \ldots K \)
    In parallel, for all \( p \in P_k \):
      Set \( \phi_p, e_p \) as per (10).
  end;
# Minimize
  In parallel, for all \( p \in P \):
    Set \( \theta_p, I_p \) as per (8) and (9).
  end;
# Downsweep
  for \( k = K, K-1, \ldots 1 \)
    In parallel, for all \( p \in P_k \):
      Set \( \theta^+_p, I^+_p \) as per (11).
  end;
  In parallel, for all \( n \):
    Set \( Z(n) \) as per (12).
end;
# Update \( \lambda^* \)
  Set \( \lambda^* = \text{MIN}(\lambda^* \times \lambda_f, \lambda) \) if \( \text{mod(iters, } T_\lambda) = 0 \).
end;
```

10