International Journal of Computer Vision manuscript No.
(will be inserted by the editor)

Correctness Prediction, Accuracy Improvement and Generalization of Stereo Matching using Supervised Learning

Aristotle Spyropoulos · Philippos Mordohai

Received: date / Accepted: date

Abstract Machine learning has been instrumental in most areas of computer vision, but has not been applied to the problem of stereo matching with similar frequency or success. In this paper, we present a supervised learning approach by defining a set of features that capture the various forms of information of each pixel, then using them to predict the correctness of stereo matches based on a random forest. We show highly competitive results in predicting the correctness of matches and in confidence estimation, which allows us to rank pixels according to the reliability of their assigned disparities. Moreover, we show how these confidence values can be used to improve the accuracy of disparity maps by integrating them with an MRF-based stereo algorithm. This is an important distinction from current literature that has mainly focused on sparsification by removing potentially erroneous disparities to generate quasi-dense disparity maps. Finally, we demonstrate domain generalization of our method by applying the classifier of a dataset to a different dataset with equally successful results.

Keywords stereo matching · 3D reconstruction · ground control points · confidence estimation

Aristotle Spyropoulos
E-mail: ASpyropo@stevens.edu

Philippos Mordohai
E-mail: Philippos.Mordohai@stevens.edu

Stevens Institute of Technology
Department of Computer Science
Hoboken, NJ, USA

1 INTRODUCTION

Stereo matching is an inverse problem and, as such, it is notoriously prone to errors, mostly due to occlusion, lack of texture and repeated structures. Since the common causes of the errors are well known, one would expect that learning methods could have been used to detect them. Helpful cues are available in the neighborhood of a pixel as well as in information generated during the matching process. Surprisingly, very few publications have attempted to tackle stereo matching from a learning perspective [8,16,17] and they have not gained much traction. In this paper we address exactly that. Given a training set of stereo pairs with ground truth disparity, the goal of this paper is to answer the following questions:

1. Is it possible to predict whether a stereo correspondence is right or wrong based on features extracted from the stereo pair for that pixel and a trained classifier?
2. Is it possible to use these predictions to improve the disparity map?
3. Can a stereo algorithm based on supervised learning generalize to a dataset different than the one it was trained on?

To answer the first question, we formulate a binary classification problem and tackle it using a random forest (RF) classifier [5]. We argue that this problem is more fundamental than confidence estimation without the ability to decide on correctness [13,26] or selection of a hypothesis among a set generated by a mixture of experts [17,20]. Ranking stereo matches accurately according to confidence is valuable, but does not imply the capability to determine which of the matches are correct, since the error rate may fluctuate from image to image making the selection of a threshold hard without knowledge of the priors. We show that we are able to predict the correctness of matches on disparity maps with very different error rates at nearly optimal levels.

To address the second question, we, first, use the results of the RF classifier to identify Ground Control Points (GCPs), that is points for which we are very confident that their calculated disparity is correct. For the identified GCPs, we modify the matching cost volume in such a way as to favor their already assigned disparity values. The modified cost volume is then used as input to a Markov Random Field (MRF) optimization that returns an improved disparity map.

We intentionally select features for our classifiers that are not domain-specific but capture general properties of successful and unsuccessful matching. In this paper, we show for the first time experiments in which a classifier is trained on a *source* dataset and then it is applied on a different *target* dataset. We demonstrate that our approach incurs minimal loss of accuracy when applied on a domain different than the one it was trained on. This type of generalization is useful in cases where ground truth depth maps may not be available for one of the datasets. It also ascertains the fact that we made every effort to keep the algorithm non-domain-specific.

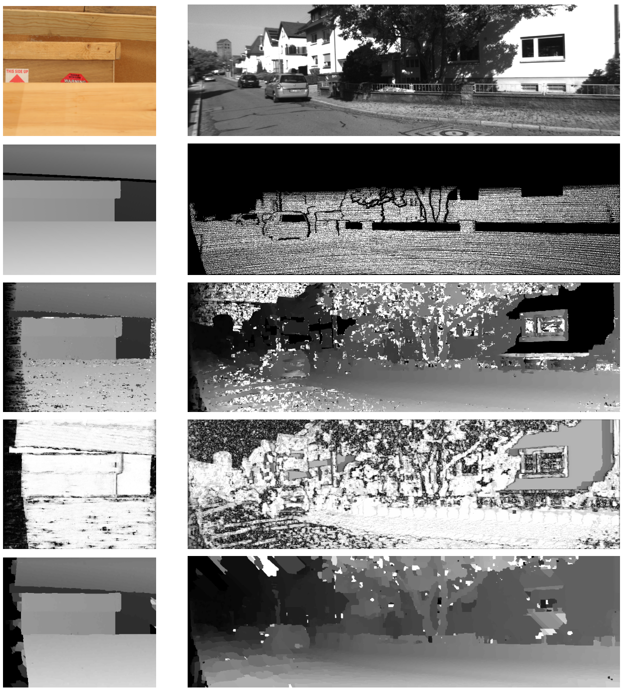


Fig. 1 Left column: Input image, ground truth, WTA disparity map using NCC for Wood2 from the Middlebury dataset [28], prediction map (bright intensities correspond to WTA matches that are likely to be correct), and final disparity after MRF optimization. Right column: Similar maps for image 102 from the KITTI dataset [9].

Our results show that we are able to successfully address all three questions. Figure 1 shows the inputs to our algorithm: the original image and a Winner-Take-All (WTA) disparity map, as well as its outputs: a correctness prediction map and an improved disparity map after MRF optimization. WTA is a local method for disparity computation, in which the disparity associated with the minimum cost value is selected at each pixel, independently of other pixels.

What separates our approach from recent literature on confidence estimation [26,10,13,27,11], regardless of the use of learning, is that the main objective of these methods is sparsification. They can indeed generate disparity maps with progressively fewer errors by removing matches starting from the least reliable ones. What has not been shown, however, is how this capability can be used to correct the initially wrong matches. Haeusler et al. [11] presented an approach for learning a confidence measure from several features, some of which are similar to those proposed by us. Haeusler et al. also use a random forest for classification, but, unlike this paper, they do not propose ways of leveraging the estimated confidence to generate dense disparity maps of higher accuracy.

Our contributions are:

- an algorithm that achieves high accuracy in predicting the correctness of stereo matching by training a classifier on stereo pairs with ground truth disparity,
- an approach for leveraging the above classifier to generate dense disparity maps of higher accuracy by detecting ground control points and for inserting them as soft constraints into an MRF-based optimizer, leading to improved disparity maps,
- a diverse set of features that enable accurate classification,
- a confidence measure that greatly outperforms all competing methods based on a recent survey [13],
- confirmation that it is feasible to apply the training results of a source dataset to improve the disparity maps of a target dataset that has completely different image characteristics.

This paper extends our previous work that was published in CVPR 2014 [30]. It includes additional details on the features used by our classifier, along with analysis of each feature on the 2006 Middlebury dataset. It shows that our algorithm can be applied on an additional (and completely different) dataset, the KITTI stereo benchmark [9], without any modifications and be equally accurate as in the Middlebury dataset. Finally, it validates our hypothesis that our classifier can be trained on a dataset and applied to a different dataset with negligible loss of accuracy.

The paper is organized as follows. Section 2 presents related work. Section 3 describes the two datasets we used in our experiments. Section 4 describes the features of our classifier in detail along with information of each feature's accuracy. Sections 5, 6 and 7 present the classifier, a method to establish ground control points and the MRF optimization, respectively. Sections 8, 9 and 10 present the results on predicting the correctness of stereo matches, ground control point accuracy and MRF optimization, including comparisons with numerous baselines. Finally, in Section 11 we show the results of applying classifiers trained on a source dataset to a different target dataset.

2 RELATED WORK

For a review of stereo methods we refer readers to surveys [29,6]. Relevant to our research is the literature on confidence estimation [26,13,10,25] from which we select features for our classifiers. Also, somewhat relevant are methods for and on learning optimization or regularization parameters [39,32,23,34] for stereo. These methods aim at learning a small number of global parameters, such as the weights of the data and smoothness terms of an MRF, while our work aims to train classifiers that make decisions per pixel. In this section, we focus on research that aims at inferring the correctness of correspondences using learning, or at detecting ground control points (GCPs).

Early work on applying machine learning to stereo includes that of Lew et al. [18] who presented an approach for selecting a set of features that form an effective descriptor for stereo matching. Cruz et al. [8] addressed the problem of determining whether a match in edge-based stereo was correct or not. Classification relies on four features extracted by filtering the images and uses a perceptron to determine which feature mappings from the left to the right image are indications of correct matching. This approach, however, does not address challenges in textureless regions, since it is only applied to edge pixels, and also does not model mismatches due to repeated structures.

Kong and Tao [16] used non-parametric techniques to learn the probability of a potential match to belong in three categories: correct, wrong due to foreground over-extension or wrong for other reasons. They used features extracted from image appearance and matching cost estimates, while final disparity assignments to fronto-parallel superpixels were made via simulated annealing on an MRF. The integration of the correctness probabilities into the MRF improved accuracy on the Middlebury benchmark, but the accuracy of the standalone classifier was not reported in the paper. This approach was extended [17] to select among 36 experts in the form of different normalized cross-correlation (NCC) matching windows using similar features and optimization technique. Motten et al. [22] presented a classifier using decision trees implemented on FPGA for selecting among multiple disparity hypotheses generated by trinocular stereo. A different approach based on a Conditional Random Field (CRF) formulation was published by Li and Huttenlocher [19]. It learns linear discriminant functions that compute the data and smoothness terms of the CRF based on discretized values of the matching cost, image gradients and disparity differences among neighboring pixels. These linear functions are learned using a structured support vector machine. Alahari et al. [1] formulated a similar learning problem using the same node and edge features as [19] and convex optimization to obtain the solution more efficiently.

We would be remiss if we did not include the work of Mac Aodha et al. [20] on optical flow, which shares some characteristics with ours, such as an emphasis on being applicable to general scenes and operating on individual pixels. A multi-class classifier that selects among four state of the art methods is used to learn the posterior of each expert being correct. The estimated posteriors are then used as confidence measures. Other recent research on confidence estimation, from which we draw inspiration and borrow features, includes the work of Reynolds et al. [26] on time-of-flight data and of Hu and Mordohai [13] on stereo. Haeusler and Klette [10] also considered several confidence measures, as well as the product of all measures, demonstrating good performance in sparsification. Pfeiffer et al. [25] integrated three confidence measures into a mid-level representation for 3D reconstruction and showed that Bayesian reasoning outperforms sparsification by thresholding.

Contrasted with methods for selecting among a set of experts, such as those of Kong and Tao for stereo [17] and Mac Aodha et al. for optical flow [20], our research addresses the more fundamental problem of verifying whether a prediction from a single expert is correct. Sabater et al. [27] introduced an a contrario approach for validating the correctness of stereo matches. The approach relies on a stochastic background model, named the a contrario model, that corresponds to the probability that the similarity score between two patches has arisen by chance and not due to true correspondence. To reduce dimensionality and capture the correlations among pixels in an image patch the a contrario model is estimated by applying PCA to the set of all patches of a given size in an image. A user-specified acceptable number of false matches determines a threshold on similarity that is used to accept meaningful matches and as a result also determines the density of the final disparity map.

The most similar prior work to ours was presented by Haeusler et al. [11] who also train a random forest to predict the correctness of the output disparities of the semi-global matching (SGM) stereo algorithm [12]. It uses a number of features computed on the images, disparity maps and matching cost volume, which aim to capture the likelihood of a disparity being incorrect. A minor difference between [11] and our approach is that some of the features are computed at multiple scales. The two major differences are the stage of processing at which the classifier is invoked and the way its predictions are used. Haeusler et al. train the random forest to detect errors after SGM optimization, that is in disparity maps with very high accuracy. The classifier's predictions are then used to sparsify the disparity maps by removing potential errors. On the other hand, we train a classifier to detect errors in low-accuracy disparity maps generated after WTA disparity assignment. The output of the classifier is used to guide global optimization and improve the accuracy of the final, dense disparity maps. (We show sparsification results to evaluate our classifier in isolation in Section 8.)

Recently, Park and Yoon [24] published an approach similar to ours which also uses a number of confidence measures as features in a random forest classifier that predicts the correctness of WTA disparities. They use the classifier predictions to modulate the data term of each pixel in SGM-based optimization. The modulation leads to 1.22% reduction in matching error on the KITTI benchmark compared to regular SGM stereo. This is slightly larger than the improvement we obtain on MRF-based stereo with the addition of ground control points on the same dataset. (See Sec. 10.) Park and Yoon re-implemented the method of [11] and ours and performed a comparison of the three methods in terms of AUC on the same data. Their method ranks first followed by ours and that of Haeusler et al. [11] in that order.

Zbontar and LeCun [37] trained a convolutional neural network (CNN) to predict whether two image patches match or not. The CNN generates matching costs which are adaptively aggregated [38] and optimized using SGM to obtain the top-ranking results on the KITTI benchmark. Zagoruyko and Komodakis [36] compared multiple CNN architectures applied to a wide range of matching problems including stereo matching. Both papers demonstrate that CNNs are more effective than manually designed descriptors and matching functions. While both the approaches of [37,36] and ours address binary classification problems, theirs take as input only the images and computes a matching cost volume, while ours takes as input the images, the matching cost volume and the WTA disparity map and predicts whether the assigned disparities are correct.

Methods for selecting GCPs typically rely on heuristics that are strongly correlated with correctness, but make hard decisions based on multiple thresholds. Bobick and Intile [3] imposed several constraints on GCPs: lower cost than all competing matches in both images, low matching cost, sufficient image texture and presence of nearby GCPs to suppress outliers. Kim et al. [14] use left-right consistency (LRC) and comparison of the matching cost against a threshold for selecting GCPs. Wang and Yang [33] pick GCPs by running three different Winner-Take-All (WTA) stereo algorithms and require that the disparities be consistent among all the matchers in each image, as well as left-right consistent. Sun et al. [31] used LRC and the ratio of the best to the second best matching cost in a disparity propagation framework.

Our approach integrates numerous criteria in a principled way via supervised learning and learns how to make decisions based on labeled data rather than intuition.

3 DATASETS

We used two datasets for our experiments:

The extended **Middlebury** stereo dataset [28] consists of six stereo pairs from the 2005 data (the remaining three do not have ground truth disparity maps available) and all 21 images from the 2006 data, for a total of 27 stereo pairs. The images were captured indoors in a lab environment and depict objects with varying complexity, as shown in Fig. 2.

We use the one third-size RGB images with resolutions varying from 413×370 to 465×370. The maximum disparity at that size is 85 with a minimum disparity of 0. As per the dataset's specifications, the values of the calculated disparities are considered correct if the difference to the true ground truth is within ±1.

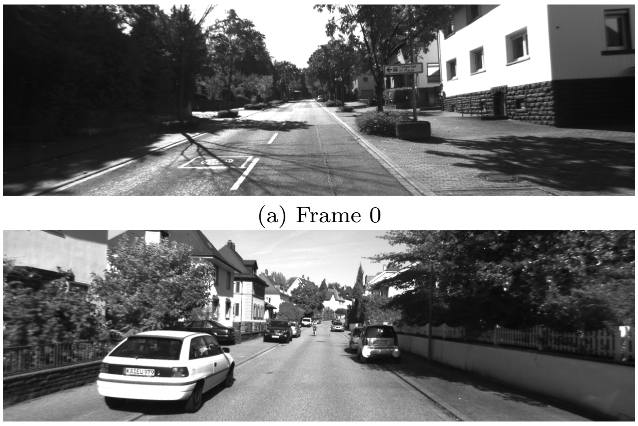
The stereo benchmark of the **KITTI** Vision Benchmark Suite [9] consists of a training set of 194 stereo pairs for which ground truth disparity maps are available, as well as a test set of 195 stereo pairs without ground truth. Since we required ground truth disparity maps for the evaluation of our algorithm, we used the 194 stereo pairs of the training set in our experiments. Images were captured by a properly equipped vehicle while driving around in rural areas and highways. Examples are shown in Fig. 3.

All images are grayscale with resolutions approximately 1240×370. The maximum disparity is 255 with

(a) Aloe (b) Cloth3

(c) Books (d) Laundry

Fig. 2 Sample images from the Middlebury dataset



(a) Frame 0

(b) Frame 6

Fig. 3 Sample images from the KITTI dataset

a minimum disparity of 0. As per the dataset's specifications, the values of the calculated disparities are considered correct if the difference to the ground truth is within ±3. However, for the KITTI dataset, due to the way the data were captured, there are no ground truth values for the top one third of each image. The average density of ground truth disparity values is 35%.

4 FEATURES

In this section, we describe the features we selected for our classifier, as well as the rationale behind them. This set of features is by no means exhaustive, but it aims at extracting useful information from various sources including the cost curve for each pixel and the pixel's neighbors in the disparity map. The label for each pixel indicates whether the disparity with the minimum cost, that would have been assigned to it by a WTA stereo algorithm, is correct or not. The definition of correctness that is appropriate for each dataset is used.

Before describing the features, we introduce some notation. Given a pair of rectified images, we compute the *cost volume* $c(x_L, x_R, y)$ that contains a cost value for each possible match from a pixel in the left image (x_L, y) to a pixel in the right image (x_R, y). Disparity is defined conventionally as $d = x_L - x_R$ and we assume that the minimum and maximum values it can take, d_{min} and d_{max}, are externally provided. For convenience, we define the disparity of a pixel in the right image to be equal to d, $d_R = x_L - x_R$. Values in the cost volume for matches beyond the disparity range are flagged as invalid and ignored in all computations. If a similarity, instead of a cost function, is used to assess matches, we negate its output to convert it to cost. The *cost curve* of a pixel is the set of cost values for all allowable disparities for the pixel. We use c_1 and c_2 for the minimum and second minimum values of the cost curve, without requiring c_2 to be a local minimum. The disparity value corresponding to c_1 is denoted by d_1.

We used the following eight features for the experiments in this paper. Four of them (MMN, AML, LRC, LRD) were considered individually as confidence measures in [13]. Note that all evaluations shown below were based on the 2006 Middlebury dataset.

4.1 Cost

The first feature is the minimum matching cost over all disparities for a given pixel and it captures the fact that low cost often corresponds to high likelihood of correct matching. We selected the negated Normalized Cross-Correlation (NCC) in a 5×5 window as the cost function. The choice of matching function and window size is not optimized in any sense, but produces reasonable results. The resulting accuracy is shown in Fig. 4.

$$f_{\text{cost}} = \frac{-\sum_{i \in W} (I_L(x_i, y_i) - \mu_L)(I_R(x_i - d, y_i) - \mu_R)}{\sigma_L \cdot \sigma_R} \tag{1}$$

where I_L and I_R are the two images of the stereo pair, μ_L and μ_R are the means and σ_L and σ_R are the standard deviations of all pixels in the square window W in the left and right image, respectively. Means are computed separately per RGB channel, but a single standard deviation is estimated for the $3 \times N \times N$ vector obtained by stacking all the elements in the window after the mean RGB values have been removed. This reduces sensitivity to image regions with small variance in any one channel. All positive cost values are truncated to 0.

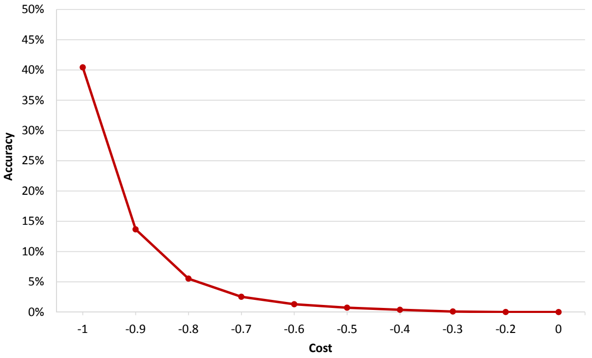


Fig. 4 Percent of pixels with correct disparity by negated NCC value (cost) in a 5x5 window. Average accuracy across all the Middlebury images is 67.2% All positive cost values are truncated to 0.

4.2 Distance from Border (DB)

This feature measures the distance in pixels from the nearest image border. It is based on the assumption that pixels near the borders are likely to be outside the field of view of the other camera and that fact causes mismatches. We also experimented with four separate features measuring the distance from the left, right, top and bottom borders, but no improvement was observed. Figure 5 displays the error rates at various distance ranges. Since pixels with a distance of less than or equal to 5 pixels from any border are most likely to be wrong, we implemented DB as a binary feature that takes a value 0 (if the distance is within the 5 pixel range), or 1 (if the distance exceeds 5 pixels).

4.3 Maximum Margin (MMN)

This feature measures the difference between the two smallest cost values, c_1 and c_2, of a pixel [13] as shown in Eq. 2. The rationale here is that a large difference may indicate an unambiguous disparity assignment.

$$C_{\mathrm{MMN}} = c_2 - c_1 \tag{2}$$

Figure 6 illustrates the MMN definition by displaying the cost of a single image pixel for all possible disparity values (here 0 to 85). The lowest cost, c_1, which has the highest probability of belonging to the correct disparity value, is observed at a disparity value of 21. The second best, c_2, appears at a disparity value of 22. The absolute difference between the two costs (approximately $0.9 - 0.6 = 0.3$) is defined as the MMN value.

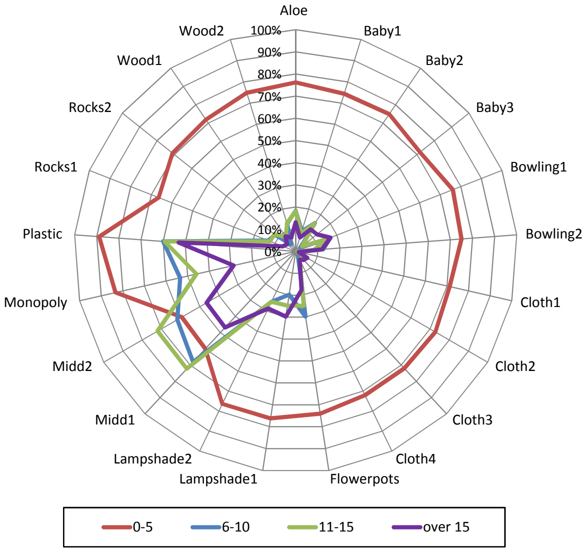


Fig. 5 Error rate by distance from border for each of the Middlebury images. The outer red line represents the error rate (in %) for pixels with a distance of up to 5 pixels away from the image's borders, an error that in most cases exceeds 70%. Areas located at distances over 5 pixels from a border exhibit significantly lower error rates, as the remaining plot lines demonstrate.

4.4 Attainable Maximum Likelihood (AML)

This feature is based on the conversion of the cost curve to a probability density function over disparity. It has been shown that subtracting the minimum cost $c_1(x_L, y)$ from all cost values leads to higher discriminative power [21, 13]. This is based on the observation

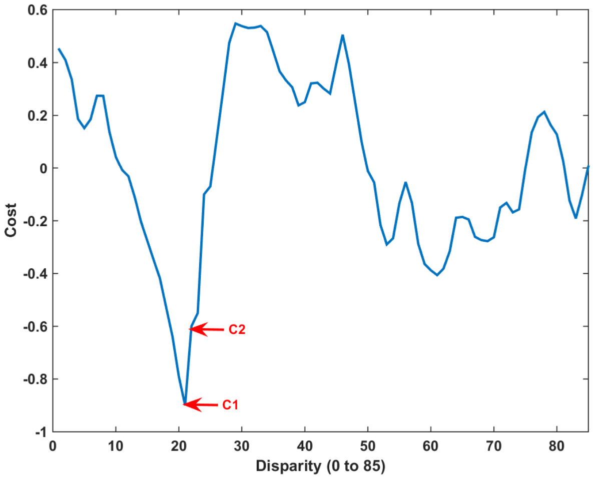


Fig. 6 The cost curve of a single pixel for each of the possible disparities (0 to 85 for Middlebury). The difference between c_1 and c_2 is defined as the Maximum Margin.

that correct matches are often associated with matching costs that are far from optimal. For these pixels, the costs of all disparities are elevated and when costs are converted to likelihoods and normalized, the resulting probability mass function appears more uniform than it actually is. This leads to pixels with unambiguous disparities that have lower confidence than more ambiguous pixels with lower minimum matching costs. Subtracting the minimum observed for a given pixel from all cost values alleviates this problem. First we use the cost curve of a pixel across all possible disparity values (Fig. 7, top) to calculate the likelihood (Fig. 7, middle) of each disparity value using Eq. 3.

$$f_{\text{Likelihood}}(x_L, y, d) = exp(-\frac{(c(x_L, y, d) - c_1(x_L, y))^2}{2\sigma^2_{AML}}) \tag{3}$$

AML models the cost for a particular pixel using a Gaussian distribution centered at the minimum cost value that is actually achieved for that pixel (c_1 in our notation). The likelihood is then normalized (Fig. 7, bottom) as follows:

$$f_{\text{AML}}(x_L, y, d) = \frac{f_{\text{Likelihood}}(x_L, y, d)}{\sum_d f_{\text{Likelihood}}(x_L, y, d)} \tag{4}$$

AML of a pixel (x_L, y) is defined as the normalized likelihood of the disparity with the minimum cost:

$$f_{\text{AML}}(x_L, y) = \frac{1}{\sum_d exp(-\frac{(c(x_L, x_R, y) - c_1(x_L, y))^2}{2\sigma^2_{AML}})} \tag{5}$$

Notice that $f_{Likelihood}$ is 1 for the disparity with the minimum cost.

4.5 Left-Right Consistency (LRC)

A good indicator of the correctness of a match from the left to the right image is whether the match is confirmed in the opposite direction. LRC, here, is a binary feature set to 1 when the absolute value of the difference between the disparity d at pixel (x_L, y) in the left image and the disparity at pixel $(x_L - d, y)$ in the right image is less than or equal to 1. LRC is 0 when the difference is greater than 1. Figure 8 depicts LRC accuracy on the Middlebury stereo pairs. The average accuracy over all stereo pairs is 78.3%.

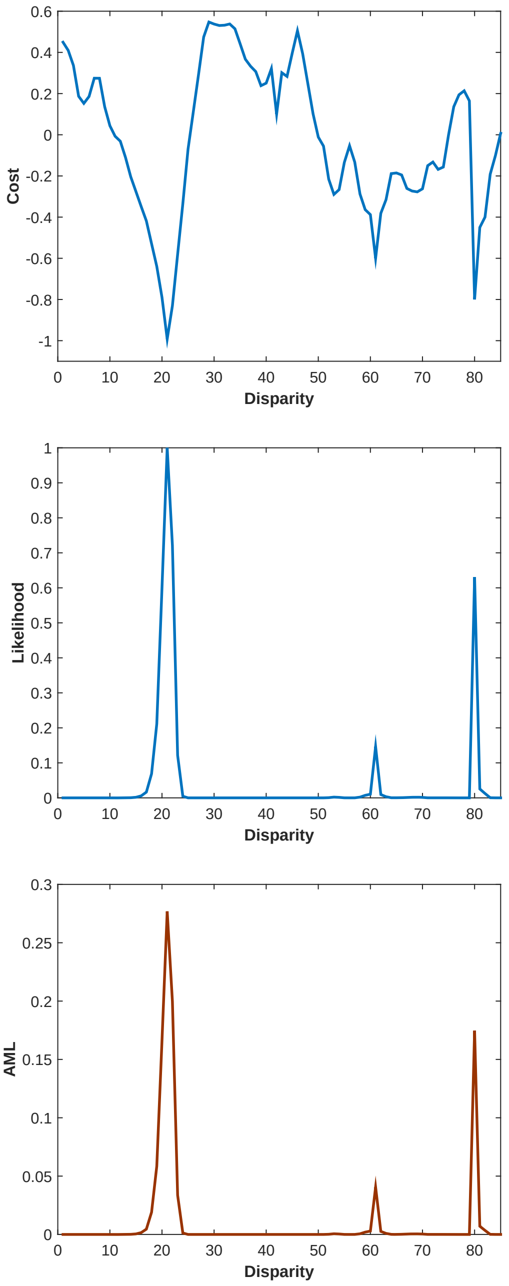


Fig. 7 Calculation of AML from cost function (top), to likelihood (middle), to AML (bottom) for a range of probable disparities of a given pixel.

4.6 Left-Right Difference (LRD)

This confidence measure [13] favors a large margin between the two smallest minima of the cost for pixel (x_L, y) in the left image and also consistency of the minimum costs between the left-to-right and right-to-left disparity maps.

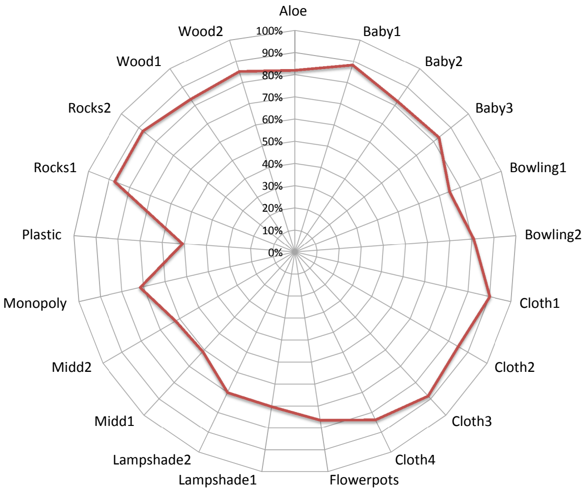


Fig. 8 Average Accuracy by stereo pair for LRC value equal to 1.

$$f_{\mathrm{LRD}}(x_L, y) = \frac{c_2(x_L, y) - c_1(x_L, y)}{|c_1(x_L, y) - min_{x'}\{c(x', x_L - d, y)\}|} \quad (6)$$

The intuition is that truly corresponding pixels should result in similar cost values and thus a small denominator. LRD can be small for two reasons: if the margin is small, or if the margin $c_2 - c_1$ is large, but the pixel has been mismatched causing the denominator to be large.

4.7 Distance from Discontinuity (DD)

Pixels near depth discontinuities are likely to be mismatched due to pixel blending. Pixels away from discontinuities show remarkable disparity accuracy. As a matter of fact, most errors occur on pixels very close to discontinuities, making this measure a significant addition to our classifier. Since we do not know the true discontinuities, we use the WTA disparity estimates as a proxy and declare as discontinuous any pixel whose disparity is not equal to all of its four neighbors. DD is then equal to the horizontal distance of each pixel to its nearest discontinuity.

4.8 Difference with Median Disparity (MED)

Pixels with disparity values that are consistent with their neighborhood are more likely to be correct. We capture this by computing the median disparity in a 5×5 window centered at each pixel and taking the absolute value of the difference between the median and the pixel's own disparity. This difference is truncated at 2 in our current implementation, as values above 2 correspond to a uniformly low accuracy. The possible values are, then, 0 (if the actual difference is 0), 1 (if the actual difference is 1), or 2 (if the actual difference is greater than 1). A plot of MED accuracy is depicted in Fig. 9.

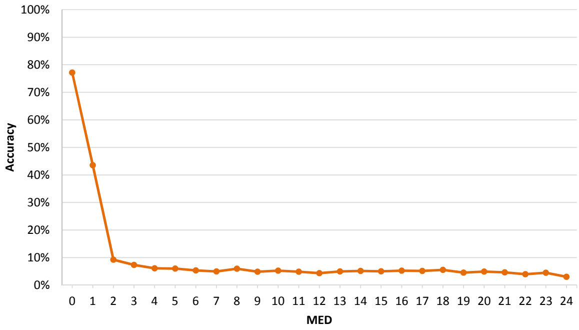


Fig. 9 Difference with Median Disparity. MED values of 0 correspond to almost 80% accuracy and MED values of 1 to about 45%. An obvious drop of accuracy is observed for values above 1.

We experimented with additional features, but none of them appeared to contribute towards higher prediction accuracy. For example, we were not able to extract useful information from image appearance using gradient or color variance-based features. We speculate that the reason is that large gradients are associated with discontinuities that have large mismatch probability, but also with highly textured pixels that can be reliably matched. We also tried a feature that indicates whether a pixel is occluded according to current disparity estimates, but it also appears to offer little additional benefit. Other features from [13] are either weak predictors or strongly correlated with the ones above.

5 CLASSIFIER

Our feature design was not done with any learning algorithm in mind, an approach that allowed us to experiment with different options. We have selected a Random Forest (RF) [5, 7] among alternatives, such as linear and nonlinear Support Vector Machines (SVM), methods that performed worse in our tests. Random forest classifiers are ensembles of classification and regression trees that have gained popularity due to their high accuracy and ability to generalize. They are well suited for inhomogeneous feature spaces, such as ours, because, unlike SVMs for example, they do not require a distance metric in feature space. Scaling the features to make them amenable to an SVM is often a daunting task. The key idea during training is to generate decision trees that partition the feature space separating the training data according to their labels, which are correct and incorrect disparities, in our case. We believe that the non-parametric nature of the random forest and its resilience to noisy labels make it a good fit for our data. Boosting, may have also been successful, but we did not attempt it.

We begin by splitting the data into a training and a test set. The training set can be viewed as a collection of pixels with assigned disparities, feature vectors and correct/incorrect labels coming from all stereo pairs. Each of the image's pixels was associated with a feature vector containing eight values (one for each of the eight features described in the previous section), as well as one binary label indicating whether the disparity assigned to that pixel was correct (1) or not (0). Occluded pixels are ignored during training. Training sets for each tree were chosen by bagging, that is by sampling from the entire training set with replacement [5]. At each node of a tree, a feature is randomly selected and a split that separates the pixels according to the correctness of their disparity, their label, is found. Splitting continues until the resulting nodes would have less than a pre-determined number of samples. Nodes that are not split are leaves of the tree. This process is repeated for each tree without any pruning.

Once the RF has been trained, the pixels of the test set with their assigned disparities and feature vectors are presented to each trained tree in the RF. The current pixel is run down each tree and decisions are made at every node based on the optimal splits computed during training. This process continues until a terminal node (leaf) is reached and a decision is made about the current pixels class label. The RF averages the predictions of the trees to assign a score between 0 and 1 that serves as a soft prediction of the correctness of each pixel. A number closer to 1 represents a high prediction accuracy. These predictions can be viewed as confidence measures. They can be used to rank disparity assignments, or they can be thresholded to classify them. Since we cannot expect to know whether a pixel is occluded during testing, we included the occluded pixels in the test set without distinguishing them from non-occluded pixels. The ground truth labels for the occluded pixels were treated identically to those of the non-occluded ones.

We estimate feature importance by measuring the increase in prediction error on a validation set if the values of that feature were permuted as in [5]. As a validation set for each tree, we use the out-of-bag samples, that is the samples not selected for the training set of that tree during the bagging process. If a feature is irrelevant for prediction, perturbing its values would lead to no change in the accuracy of each tree and the random forest. On the contrary, if a feature is important, perturbing its values leads to increases in prediction error on the out-of-bag samples. Table 1 reports feature importance for an RF trained on the Middlebury data. The values shown are increases in prediction error averaged over all trees and normalized by the standard deviation over the entire random forest. While there is some variability among different RFs trained on the same data due to the randomness of the procedure, the ordering and approximate magnitude of the importance values is stable.

As expected, not all features are equally important to the classifier. Six features have a very similar importance with values around 1.0, while two (DB and AML) have lower importance. DB is relevant for a small fraction of pixels near image borders.

We further experimented with an RF that uses only five of the eight features by removing DB, AML, and LRD, since their RF classifier importance values were lower than those of the other five features. Although the error rate for the Middlebury dataset was about the same as when using eight features, in the KITTI dataset there was an increase of the error rate by 0.10%. This demonstrates that despite their lower importance value, additional features can be useful by providing some redundancy and stability to the classifier.

6 GROUND CONTROL POINT SELECTION

Having calculated a prediction accuracy for each pixel, we present next an approach for selecting ground control points (GCPs). The GCPs are used in the next section to improve WTA disparity maps via global optimization. Consistent with earlier definitions [33] [14], a GCP is defined here as a pixel with a disparity assignment that is assumed to be very reliable and, therefore, can be used to influence neighboring pixels. The goal is to achieve the highest possible density of GCPs while

#	Feature	Importance
1	Cost	0.9707
2	DB	0.4102
3	DD	1.4169
4	LRC	1.1404
5	MED	1.0872
6	MMN	0.9274
7	AML	0.5857
8	LRD	0.8365

Table 1 RF classifier importance values for each of the eight features in the Middlebury dataset. See text for details.

including a small number of wrong matches. If GCPs are not accurate and contain many pixels with wrong disparities, these errors will be propagated to neighboring pixels and can have a strong negative effect on overall accuracy. On the other hand, if GCP detection is overly conservative, the small number of selected GCPs has little effect on overall accuracy, since they do not appear in uncertain regions of the images.

Since the random forest has proven very effective in ranking disparity assignments in order of reliability, we chose GCPs by learning a threshold on the RF prediction that resulted in the highest overall disparity accuracy after MRF optimization, as shown in Section 10. The threshold was learned using binary search on a range of RF values from 0.50 to 0.95. However, we chose not to impose them as hard constraints. Among several alternatives, we decided on the following that was proven to be superior experimentally: when the random forest predicted that a given disparity assignment to a pixel is reliable, we set the cost of all other disparities for the pixel to a constant value c_{GCP}, leaving the cost for the selected disparity unchanged. This allowed the MRF to override the GCPs at a higher cost. Figure 10 depicts the transformation of the original cost values of a GCP to the cost values that were used as input to MRF. The cost of all disparities of non-GCPs remained unchanged in the [-1, 1] range of negated NCC.

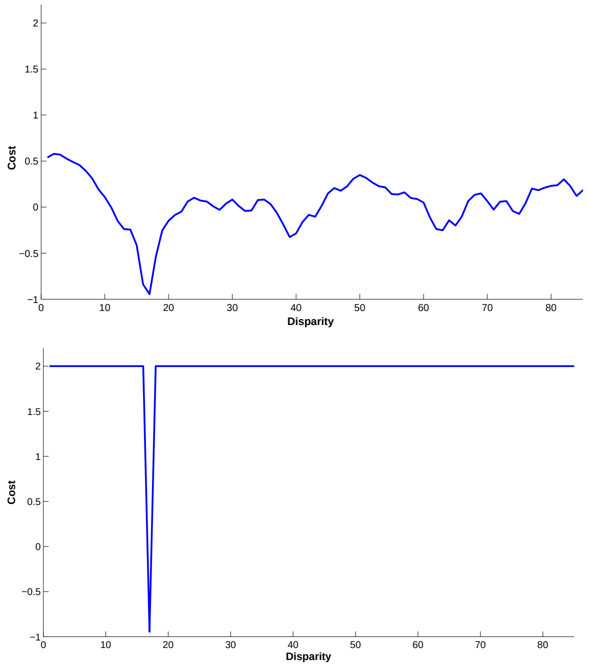


Fig. 10 Cost curve: Original cost curve (top) of a single GCP across all possible disparity values and as modified for MRF input (bottom).

7 GLOBALLY OPTIMIZED DISPARITY MAPS USING GCPs

With the selection of the GCPs and the modification of the cost curves of each GCP as described in the previous section and in Fig. 10, we proceeded to use this modified cost as input to 4-connected MRF. The MRF minimizes an energy function with data and smoothness terms of the disparity map D as follows:

$$E(D) = E_{data}(D) + E_{smooth}(D) \tag{7}$$

where:

$$E_{data}(D) = f_{cost}(D) \tag{8}$$

with $f_{cost}(D)$ as defined in Eq. 1. E_{smooth} follows a simple Potts model with edge weights modulated by the strength of the intensity edges between neighboring pixels. The smoothness energy is defined as:

$$E_{smooth}(D) = \lambda \sum_{p \in I_L} \sum_{q \in N_4(p)} \omega_{pq}[d_p \neq d_q], \tag{9}$$

where p is a pixel in the left image I_L with disparity d_p, q is a pixel in p's neighborhood with disparity d_q, λ is a parameter. The edge weights, partially adopting the settings of Wang and Yang [33], are defined as:

$$\omega_{pq} = max\{e^{-\frac{\Delta c_{pq}}{\gamma_c}}, 0.0003\}, \tag{10}$$

with Δc_{pq} the Euclidean distance of the RGB values of p and q, and γ_c equal to 3.6. These settings are constant regardless of how the GCPs were chosen.

We use the Fast-PD optimization algorithm of Komodakis et al. [15] to generate the final disparity maps given these energy functions as inputs. As all multi-label MRF optimization algorithms relying on graph-cuts, Fast-PD solves a number of max-flow problems on a series of graphs but in addition to the original, primal MRF problem, it also considers its dual. Intermediate solutions of the dual problem enable the algorithm to reduce the complexity of the max-flow problem solved at each iteration until convergence is achieved. Fast-PD guarantees the same solution as the α-expansion algorithm, which can be obtained substantially faster, and it also guarantees an almost optimal solution for a much wider class of NP-hard MRF problems. The former property is important for the types of problems encountered in this paper, since the energy function is of the same form as the one in the seminal work of Boykov et al. [4]. Their α-expansion algorithm could have been used to produce the same disparity maps at the expense of additional computational time. The resulting optimized disparity maps are shown in Section 10.

8 EXPERIMENTAL VALIDATION OF PREDICTION ACCURACY

In this section, we present results that show the ability of our approach to classify and rank matches without modifying them. The output of WTA stereo is used as-is in this section.

All experiments were performed on cost volumes computed using normalized cross-correlation (NCC) in 5×5 windows and negating the NCC values to obtain costs. The window size was selected arbitrarily because it achieves reasonably accurate WTA disparity maps. σ_{AML} in Eq. 5 was set to 0.2. We trained random forests comprising 50 trees using the Matlab TreeBagger package. Each forest has 50 trees, since increasing this value did not improve performance. All parameters were set to their default values, except for the minimum number of samples per leaf which was set to 5,000 and the number of variables from which to sample a weak learner which was set to 1. These values were set empirically using validation data to test that overfitting was avoided. The resulting decision trees have average depth equal to 12.5 for the Middlebury data and 17 for the KITTI data. The maximal depths are 18 and 21 respectively. The larger trees for KITTI are due to the availability of more training data.

It is important to distinguish between *disparity errors*, which are defined as pixels with incorrect disparities, and *prediction errors*, which are errors made by our classifier by considering a disparity assignment as incorrect, when it was correct and vice versa.

Following recent publications on evaluating the confidence of stereo [13], time-of-flight data [26] and optical flow [20], we evaluated the accuracy of the ranking of disparity assignments using receiver operating characteristic (ROC) curves of error rate as a function of disparity map density. We ranked all matches in decreasing order of prediction and produced disparity maps of increasing density by selecting pixels according to rank. The area under the curve (AUC) quantifies the ability of a confidence measure to predict correct matches. Better confidence measures result in lower AUC values. The optimal AUC can be achieved by selecting all correct disparities before starting to fill the quasi-dense disparity maps with the remaining wrong ones. As shown in Eq. 11, the optimal AUC is given by:

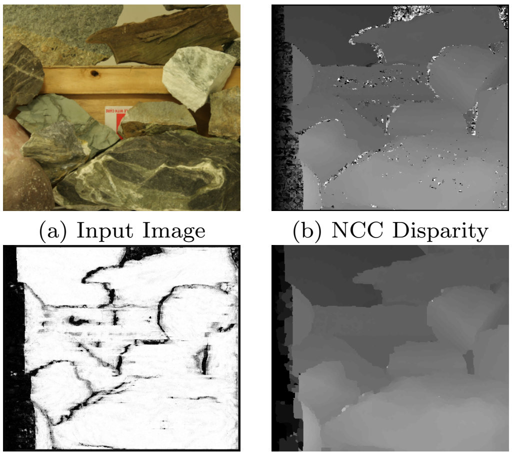

(a) Input Image (b) NCC Disparity
(c) Prediction Map (d) MRF Disparity

Fig. 11 Middlebury dataset: Input image, disparity map using NCC, prediction map and final disparity map using MRF optimization for Rocks1. Notice the low predictions (dark pixels) for occluded regions and other errors. The error rates for NCC and MRF were 5.9% and 2.6% respectively.

$$A_{opt} = \int_{1-\varepsilon}^{1} \frac{d_m - (1-\varepsilon)}{d_m} dd_m = \varepsilon + (1-\varepsilon) ln(1-\varepsilon) \quad (11)$$

where ε is the disparity error rate [13].

The results of our experimentations on each dataset are presented in the following sub-sections. All errors shown are test errors.

8.1 Middlebury Dataset

Three-fold cross-validation was used by training a random forest on 18 stereo pairs and testing on the 9 remaining pairs. A stereo pair is always tested using the random forest that did not consider it during training. Figure 11 contains an example that shows the ability of the RF to assign low prediction scores to unreliable pixels.

In Table 2, we report number of pixels with correct disparity that were found to be correct by our classifier. Similarly, we report the number of pixels with incorrect disparity that were found to be incorrect by our classifier on the 27 stereo pairs.

We classify disparity assignments of WTA stereo by thresholding the prediction of the random forest at 0.5. Our method is effective for disparity maps with both low and high error rates. Low sensitivity to input variability differentiates our work from confidence estimation methods which may be able to rank matches accurately, but are unable to determine which ones are correct without knowledge of the disparity error rate.

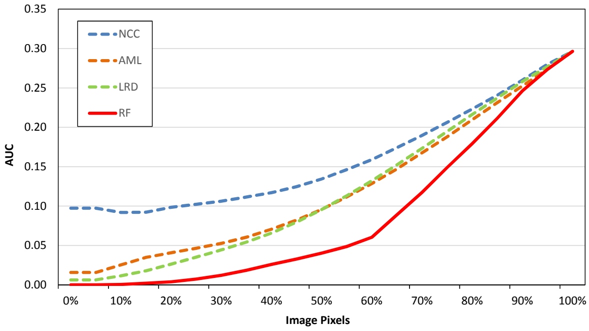


Fig. 12 Middlebury dataset: AUC values for Bowling1 at various intervals. Curves for NCC, AML, LRD and the RF prediction are displayed. The RF curve (solid red curve) has the lowest value at every point of the curve.

The overall prediction error for pixels with correct disparity is 4.72% and for pixels with incorrect disparity it is 16.01%, for a combined prediction error of 7.2%.

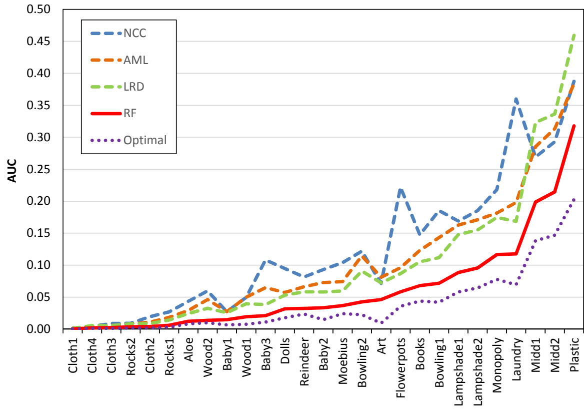


Fig. 13 Middlebury dataset: AUC values obtained by sorting the disparity assignments according to NCC, AML, LRD and the RF prediction (solid red curve) in comparison to the optimal curve (dotted line). The optimal curve is obtained with perfect knowledge of the correct matches. Please see Eq. 11 and text for details. Disparity maps have been sorted in order of increasing RF AUC to aid visualization.

	Correct Disparity	Incorrect Disparity
Total Pixels	2,687,850	670,333
Prediction Accuracy	95.28%	83.99%

Table 2 Middlebury dataset: Total number of pixels with correct/incorrect disparity that were found to be correct/incorrect by our classifier on WTA disparity assignments for non-occluded pixels over all 27 stereo pairs. Prediction was thresholded at 0.5. *The overall accuracy of the classifier is 92.8%.*

Method	NCC	AML	LRD	RF	Optimal
AUC	0.1245	0.1034	0.0987	**0.0618**	0.0386

Table 3 Middlebury dataset: AUC values for the various methods

Figure 12 shows the AUC values obtained for a single image and at various pixel densities for NCC, AML, LRD and the RF prediction. Figure 13, on the other hand, shows the total AUC values obtained by each method for all images in comparison to the optimal curve (dotted line). Our method achieves the minimum AUC for every Middlebury stereo pair.

Table 3 shows the corresponding numeric AUC values for each method. Our method (RF) has an average AUC that is roughly one half of that of the baseline methods. Is is also superior to all other methods on every stereo pair. In fact, the average AUC generated by our method is closer to the optimal average AUC than that of the second best method (LRD).

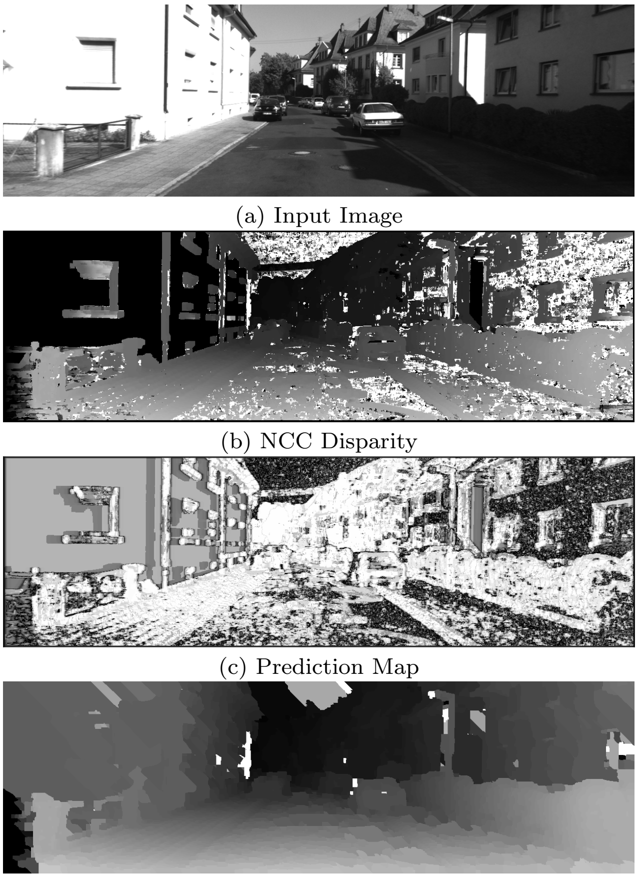
8.2 KITTI Dataset

For the KITTI dataset, we split the available stereo pairs equally in training and test sets. We trained a random forest on 97 stereo pairs (images 0 to 96) and tested on the 97 remaining pairs (images 97 to 193). Figure 14 displays a similarly noisy example from the KITTI dataset to emphasize the ability of the RF to assign low prediction scores to unreliable pixels.

In Table 4, we report the total number of pixels with correct/incorrect disparity that were found to be correct/incorrect by our classifier on the 97 stereo pairs. Similarly to Middlebury, we classify disparity assignments of WTA stereo by thresholding the prediction of the random forest at 0.5. The overall prediction error for pixels with correct disparity is 14.3% and for pixels with incorrect disparity it is 11.6%, for a combined prediction error of 13%.

	Correct Disparity	Incorrect Disparity
Total Pixels	5,513,918	5,192,949
Prediction Accuracy	85.72%	88.39%

Table 4 KITTI dataset: Total number of pixels with correct/incorrect disparity that were found to be correct/incorrect by our classifier on WTA disparity assignments for non-occluded pixels over the 97 stereo pairs. Prediction was thresholded at 0.5. *The overall accuracy of the classifier is 87%.*



(a) Input Image

(b) NCC Disparity

(c) Prediction Map

(d) RF-GCP MRF Disparity

Fig. 14 KITTI dataset: Input image, disparity map using NCC, prediction map and final disparity map using MRF optimization for 000105. Disparity maps were enhanced to aid in visualization.

Figure 15 shows the AUC obtained by each method for the complete KITTI test set (97 stereo pairs). *Our method achieves the minimum AUC for every stereo pair in the KITTI dataset as well.* Moreover, its average AUC is closer to the optimal one than to that of the best baseline method. Table 5 shows the corresponding numeric AUC values for each method.

Method	NCC	AML	LRD	RF	Optimal
AUC	0.4919	0.3360	0.3074	**0.1896**	0.1248

Table 5 KITTI dataset: AUC values for the various methods

9 GROUND CONTROL POINT ACCURACY

As discussed in Section 6, our goal was to achieve the highest number of GCPs (Density) while including as many correct matches as possible (Accuracy). We define *Density* as the number of pixels we selected as GCPs versus the total number of pixels. We define *Accuracy* as the number of correct GCPs (pixels with disparity within the appropriate threshold) over the number of GCPs that our method selected. In both definitions, we only counted non-occluded pixels and pixels with available ground truth disparity. We determined thresholds for both the Middlebury and KITTI datasets. However, sensitivity to the threshold was very low as we established during our experiments. Table 6 displays the optimum Accuracy and Density values for both datasets.

We compared the GCPs selected by our approach with several alternatives, both in terms of accuracy and density of the GCPs and in terms of accuracy of the resulting, MRF-optimized, disparity maps. Through experimentation, we also established thresholds for each of the alternatives. Our results show that the RF predictions are superior in terms of final disparity map accuracy, but also in terms of GCP accuracy. In fact, the very small fraction of errors in the GCPs is what enables our method to outperform the baselines after MRF optimization. For example, on the Middlebury dataset the density of GCPs was above 90% for the easy Cloth images and below 50% for harder images, such as Midd1, Midd2 and Plastic. The KITTI dataset density was much lower ranging from 8.8% to 47.2%, but the accuracy remained at a hight 97.56%.

Our method was successful in addressing a major challenge in GCP selection: on one hand, stereo pairs, for which WTA stereo works well, often have their accuracy degraded by regularization which may over-smooth details, while, on the other hand, stereo pairs for which WTA stereo performs poorly require more regularization and small GCP sets to avoid including errors in them. The RF scores are more flexible in automatically adapting to the inherent difficulty of each stereo pair. Baseline methods lack this flexibility. The large difference in GCP density between the two datasets (see Table 6) illustrates the adaptability of our method. KITTI images contain large texture-less regions, such as walls, sky and oftentimes the road (see Figs. 1 and 17). Placing GCPs in these regions would be detrimental to the overall accuracy since disparities in these regions are most likely incorrect. Without any intervention from

Dataset	GCP Selection	Accuracy	Density
Middlebury	RF	96.5 %	72.6 %
KITTI	RF	97.6 %	25.3 %

Table 6 Average Accuracy and Density of GCPs (GCPs are pixels with an RF value > 0.70 for the Middlebury and > 0.82 for the KITTI dataset).

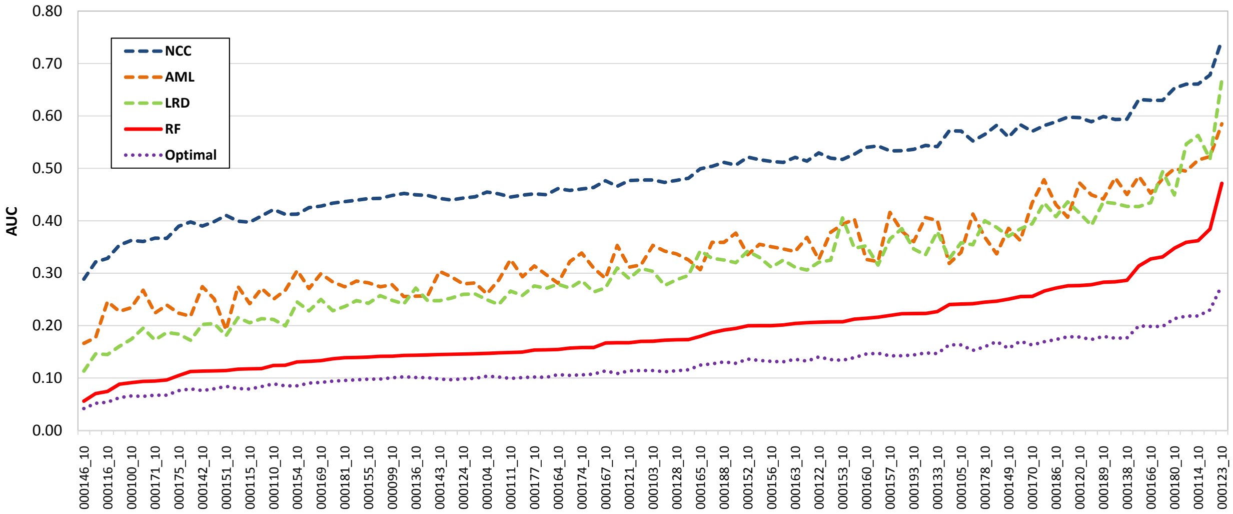


Fig. 15 KITTI dataset: AUC values obtained by sorting the disparity assignments according to NCC, AML, LRD and the RF prediction (solid red curve) in comparison to the optimal curve (dotted line). The optimal curve is obtained with perfect knowledge of the correct matches. Please see Eq. 11 and text for details. Disparity maps have been sorted in order of increasing RF AUC to aid visualization.

the user, our GCP selection technique stays away from these parts of the images.

10 MRF RESULTS

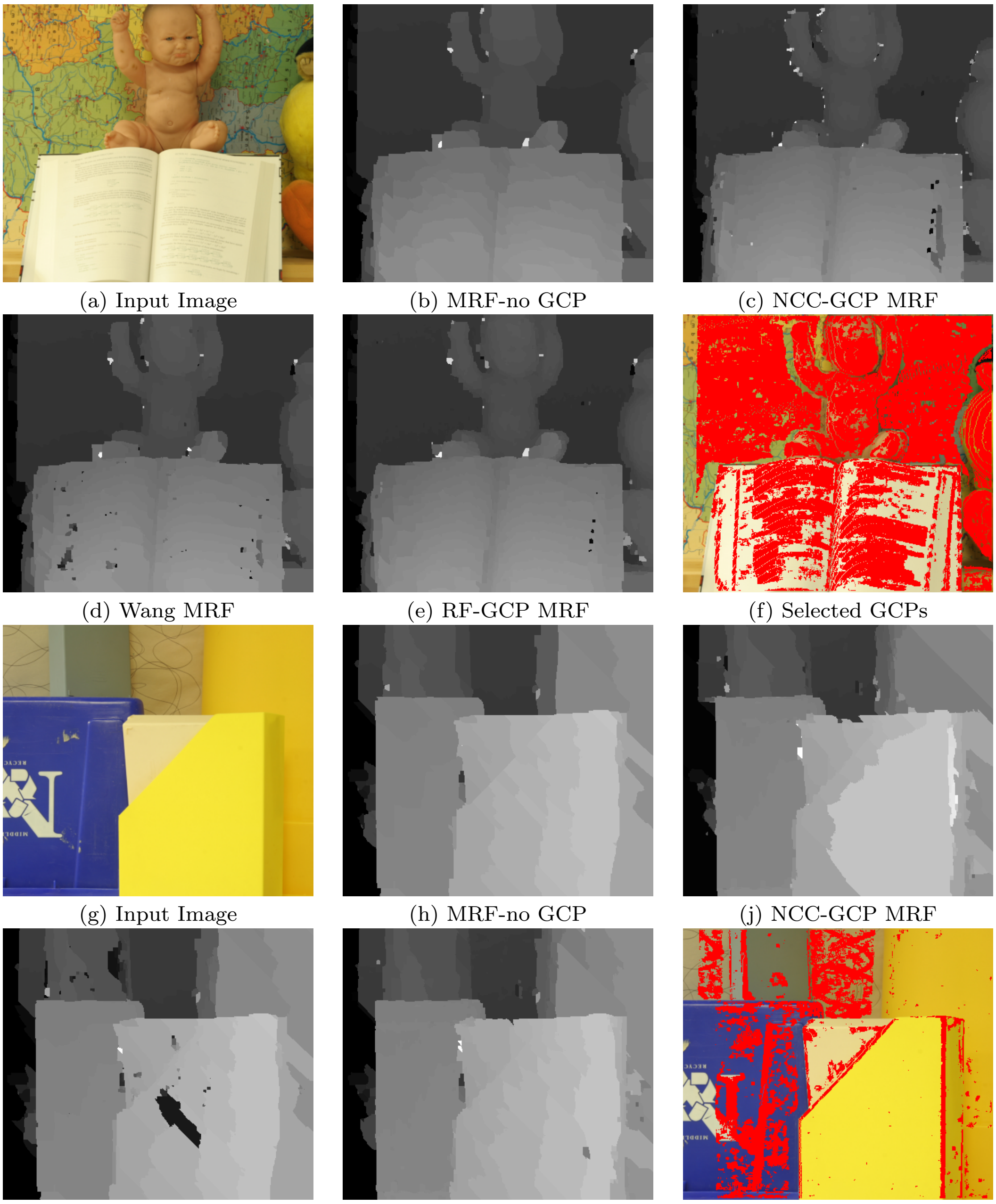
Having experimentally established a prediction accuracy threshold for our GCP selections and a method to modify the costs of the GCPs, we have now new cost values that can be used as input to the MRF optimizer. The MRF implementation of Komodakis et al. [15] was used in our experiments and the following subsections present the results on both datasets. The value of c_{GCP} was set to 2, which is twice the maximum cost that could have been observed for regular pixels.

10.1 Middlebury Dataset

We compared our method (RF) to five MRF baselines: without GCPs, using NCC, LRC or LRD values to select GCPs and finally using the algorithm of Wang and Yang [33]. The values for c_{GCP}, where applicable, λ and the threshold for each method were learned via cross-validation on the final disparity maps after global optimization. As discussed, sensitivity to the parameters has been low, as changing the RF prediction threshold from 0.7 to 0.6 resulted in an average error rate of 6.285% instead of 6.289%. Pixels were chosen as GCPs if NCC>0.5, LRC=1, LRD>1.5, or RF>0.7, respectively.

We re-implemented the method of Wang and Yang [33] which requires the agreement of three matching functions in both the left-to-right and the right-to-left disparity map for a pixel to be considered a GCP. Specifically, the three matching functions are NCC in 5×5 windows, the Birchfield and Tomasi dissimilarity measure [2] without any aggregation and the adaptive support weight method of [35] in 39×39 windows. Pixels are retained if the variance of the disparities that are estimated independently by the three methods is at most one. This test is applied on both left-to-right and right-to-left disparity maps. Pixels that pass the test in one disparity map but are not left-right consistent are rejected. Finally, pixels that are within one pixel of intensity edges detected by the Canny edge detector or have disparities at the two extremes of the disparity range are also precluded from being GCPs. The density of GCPs that survive this sequence on tests is 15.6%, which is similar to the results presented in [33] on Version 2 of the Middlebury benchmark. Conveniently, Wang and Yang used NCC in 5×5 windows as one of their matching functions. This allowed us to define the data term for regular pixels and GCPs detected by their method in a way that is identical to ours. Optimization of the resulting energy function is performed as described in Section 7.

We also tested a variant of our method that uses the GCPs detected by the RF as hard constraints. This is implemented by setting c_{GCP}, the cost of overriding the disparity of a GCP, to 100,000. Table 7 presents the error rates of the final disparity maps after MRF op-



(a) Input Image (b) MRF-no GCP (c) NCC-GCP MRF

(d) Wang MRF (e) RF-GCP MRF (f) Selected GCPs

(g) Input Image (h) MRF-no GCP (j) NCC-GCP MRF

(k) Wang MRF (m) RF-GCP MRF (n) Selected GCPs

Fig. 16 Middlebury dataset. Top: Input image and final disparity maps using an MRF without GCPs, MRFs with GCPs determined according to NCC prediction, the Wang et al. method, and RF prediction for Baby2. The last image depicts (in red) the GCPs selected using the RF prediction values. Only non-occluded GCPs and GCPs with available ground truth disparity are shown. The corresponding error rates were: (b) 8.4%, (c) 5.7%, (d) 6.2%, and (e) 3.6%.
Bottom: Similar disparity maps and RF prediction values for Plastic. The corresponding error rates were: (h) 19.2%, (j) 36.5%, (k) 21.5%, and (m) 15.8%. Despite the low GCP density, RF was able to improve the no-GCP disparity map where others failed.

timization. Our method has an improvement of 21.3% over the second best method (Wang MRF). Representative disparity maps are shown in Fig. 16.

GCP type	Average Error
WTA	22.0%
MRF-no GCP	9.8%
NCC-GCP MRF	10.0%
LRC-GCP MRF	10.3%
LRD-GCP MRF	8.7%
Wang MRF	8.0%
RF-GCP MRF	**6.3%**
RF-GCP MRF/hard	6.6%

Table 7 Middlebury dataset: The first row shows the WTA error rate. The second row shows the error rate for the plain MRF where no GCPs were selected. The next five rows show error rates after MRF optimization with pixels chosen as GCPs if NCC>0.5, LRC=1, LRD>1.5, using the method of [33] or RF>0.7, respectively. The last row shows the error rate when all GCPs with RF>0.7 were considered as hard constraints in the MRF.

10.2 KITTI Dataset

Similarly to the Middlebury dataset, using the KITTI training set we established thresholds for our method (RF), as well as the baselines. GCPs were chosen if NCC>0.5, LRC=1, LRD>15, or RF>0.82. Table 8 presents the relative error rates of the final disparity maps for the KITTI dataset. Representative disparity maps are shown in Fig. 17.

Finally, we experimented with treating the GCPs as hard constraints by setting the cost of all disparities, other than of the selected disparity, to 100,000, hence ensuring that the selected disparity would not be altered by MRF. This resulted to an increase of the error rate by 0.3% on the Middlebury dataset and by 6.6% on the KITTI dataset.

11 GENERALIZATION

In the previous sections we outlined a process that allows us to predict whether a disparity is correct, detect GCPs and, subsequently, use them to improve the accuracy of stereo matching. We have shown experimentally that in both cases (Middlebury and KITTI datasets) we have produced improved results. In this section we demonstrate that the results of learning on a domain can be applied to other domains. In the following experiments, we used the classifier obtained from a source dataset to test the accuracy on a target dataset. Moreover, the parameters c_{GCP} and λ that were established during training and testing on the source dataset were maintained intact in the target dataset. No information or training data of the target dataset were used during these experiments.

GCP type	Average Error
WTA	48.1%
MRF-no GCP	11.0%
NCC-GCP MRF	14.7%
LRC-GCP MRF	16.5%
LRD-GCP MRF	11.0%
RF-GCP MRF	**10.5%**
RF-GCP MRF/hard	17.1%

Table 8 KITTI dataset: The first row shows the WTA error rate. The second row shows the error rate for the plain MRF where no GCPs were selected. The next four rows show error rates after MRF optimization with pixels chosen as GCPs if NCC>0.5, LRC=1, LRD>15, or RF>0.82, respectively. The last row shows the error rate when all GCPs were considered as hard constraints in the MRF.

11.1 Middlebury Dataset

We used the features outlined in Section 4 to train on the 97 stereo pairs of the KITTI dataset. We then used the results of the RF to test on all 27 stereo pairs of the Middlebury dataset. Table 9 shows a comparison of the accuracy of the two test results: a) training and testing on Middlebury, and b) training on KITTI and testing on Middlebury.

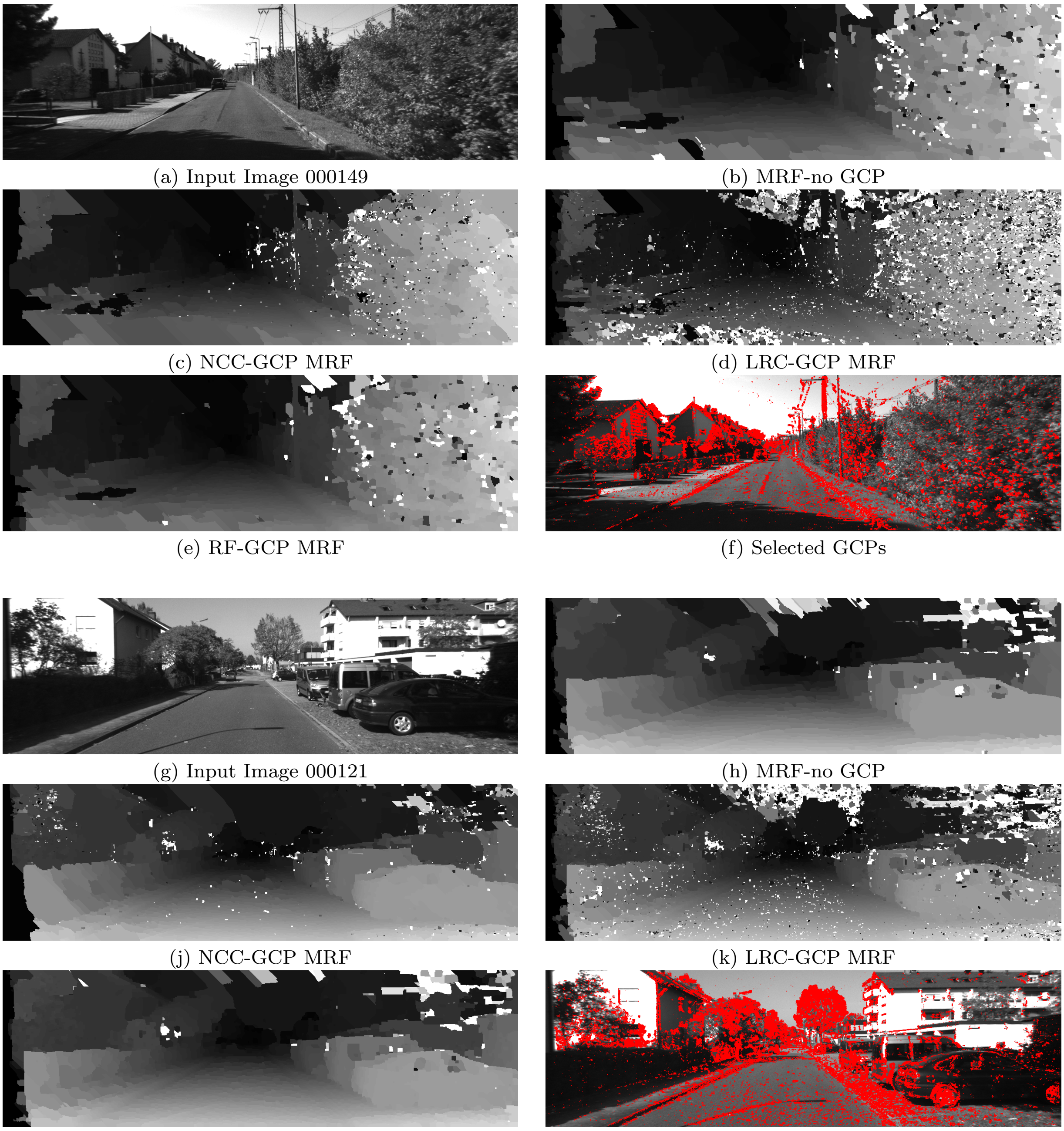
	MB → MB	KITTI → MB
Prediction Accuracy	92.79%	90.26%
MRF Error	6.29%	6.55%

Table 9 Middlebury: Comparison of Prediction Accuracy and final MRF Error using our method to train and test on Middlebury in comparison to training on KITTI and testing on Middlebury.

11.2 KITTI Dataset

Similarly, we used the features outlined in Section 4 to train on the 18 stereo sets of the Middlebury dataset. Then, we applied the results of the RF to the KITTI dataset. Table 10 shows a comparison of the two test results. Although the Middlebury images have completely different characteristics, the error rate remained practically the same.

As the experiments attest, despite the fundamental differences of the datasets, we are able to generalize our approach across domains. Using the exact same



(a) Input Image 000149

(b) MRF-no GCP

(c) NCC-GCP MRF

(d) LRC-GCP MRF

(e) RF-GCP MRF

(f) Selected GCPs

(g) Input Image 000121

(h) MRF-no GCP

(j) NCC-GCP MRF

(k) LRC-GCP MRF

(m) RF-GCP MRF

(n) Selected GCPs

Fig. 17 KITTI dataset: Top three rows: Input image and final disparity maps using an MRF without GCPs, MRFs with GCPs determined according to NCC, LRC and RF predictions for image 149. The last image depicts (in red) the GCPs selected using the RF prediction values. Only non-occluded GCPs and GCPs with available ground truth disparity are shown. The corresponding error rates were: (b) 17.1%, (c) 22.6%, (d) 24.4%, and (e) 16.3%. Bottom three rows: Similar disparity maps and RF prediction values for image 121. The corresponding error rates were: (h) 3.8%, (j) 6.4%, (k) 8.6%, and (m) 3.7%. Disparity maps were enhanced to aid in visualization.

	KITTI → KITTI	MB → KITTI
Prediction Accuracy	86.99%	81.73%
MRF Error	10.50%	10.70%

Table 10 KITTI: Comparison of Prediction Accuracy and final MRF Error using our method to train and test on KITTI in comparison to training on Middlebury and testing on KITTI.

features on both datasets, we were able to generate error rates comparable to those reported in Section 10, namely 6.3% on the Middlebury dataset and 10.5% on the KITTI dataset.

12 CONCLUSIONS

We have presented a supervised learning approach that is able to classify and rank stereo matches according to the likelihood of being correct. Experiments on standard data with ground truth demonstrate high classification accuracy, as well as ranking accuracy that is much closer to being optimal than any single confidence measure in isolation.

We have also presented a stereo algorithm that builds upon the aforementioned capabilities and global optimization techniques to improve disparity estimation accuracy. To our knowledge, these are the first results that show that disparity maps can be improved using confidence. Being able to achieve the right balance between density and accuracy of the GCPs and their use as soft constraints are important factors in the overall accuracy of our final disparity maps.

Finally, we have shown that the supervised learning approach is dataset agnostic, as the training results of a dataset can easily be applied to other datasets without loss of accuracy. In the case of applying learning from Middlebury to KITTI, we demonstrated a difference of only 0.20% in the MRF error. In the case of applying learning from KITTI to Middlebury the MRF error difference was also a low 0.26%. This is an important finding that can potentially be applied in domains such as driver assistance or autonomous driving. Applications in these domains would benefit from learning-based stereo algorithms that have good generalization properties.

ACKNOWLEDGEMENTS

The authors are grateful to Haeusler et al. [11], Park and Yoon [24] and Wang and Yang [33] for sharing data and providing guidance on how to implement their algorithms. This research has been supported in part by the National Science Foundation award #1217797.

References

1. Alahari, K., Russell, C., Torr, P.: Efficient piecewise learning for conditional random fields. In: CVPR, pp. 895–901 (2010)
2. Birchfield, S., Tomasi, C.: A pixel dissimilarity measure that is insensitive to image sampling. PAMI **20**(4), 401–406 (1998)
3. Bobick, A., Intille, S.: Large occlusion stereo. IJCV **33**(3), 1–20 (1999)
4. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. PAMI **23**(11), 1222–1239 (2001)
5. Breiman, L.: Random forests. Machine Learning Journal **45**, 5–32 (2001)
6. Brown, M., Burschka, D., Hager, G.: Advances in computational stereo. PAMI **25**(8), 993–1008 (2003)
7. Criminisi, A., Shotton, J.: Decision forests for computer vision and medical image analysis. Springer (2013)
8. Cruz, J., Pajares, G., Aranda, J., Vindel, J.: Stereo matching technique based on the perceptron criterion function. Pattern Recognition Letters **16**(9), 933 – 944 (1995)
9. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. International Journal of Robotics Research (IJRR) (2013)
10. Haeusler, R., Klette, R.: Analysis of kitti data for stereo analysis with stereo confidence measures. In: ECCV Workshops, pp. II: 158–167 (2012)
11. Haeusler, R., Nair, R., Kondermann, D.: Ensemble learning for confidence measures in stereo vision. In: CVPR (2013)
12. Hirschmüller, H.: Stereo processing by semiglobal matching and mutual information. PAMI **30**(2), 328–341 (2008)
13. Hu, X., Mordohai, P.: A quantitative evaluation of confidence measures for stereo vision. PAMI **34**(11), 2121–2133 (2012)
14. Kim, J.C., Lee, K.M., Choi, B.T., Lee, S.U.: A dense stereo matching using two-pass dynamic programming with generalized ground control points. In: CVPR, pp. 1075–1082 (2005)
15. Komodakis, N., Tziritas, G., Paragios, N.: Fast, approximately optimal solutions for single and dynamic MRFs. In: CVPR (2007)
16. Kong, D., Tao, H.: A method for learning matching errors for stereo computation. In: BMVC (2004)
17. Kong, D., Tao, H.: Stereo matching via learning multiple experts behaviors. In: BMVC (2006)
18. Lew, M., Huang, T., Wong, K.: Learning and feature selection in stereo matching. PAMI **16**(9), 869 –881 (1994)
19. Li, Y., Huttenlocher, D.: Learning for stereo vision using the structured support vector machine. In: CVPR (2008)
20. Mac Aodha, O., Humayun, A., Pollefeys, M., Brostow, G.J.: Learning a confidence measure for optical flow. PAMI **35**(5), 1107–1120 (2012)
21. Merrell, P., Akbarzadeh, A., Wang, L., Mordohai, P., Frahm, J.M., Yang, R., Nistér, D., Pollefeys, M.: Real-time visibility-based fusion of depth maps. In: ICCV (2007)
22. Motten, A., Claesen, L., Pan, Y.: Trinocular disparity processor using a hierarchic classification structure. In: IEEE/IFIP International Conference on VLSI and System-on-Chip (2012)
23. Pal, C., Weinman, J., Tran, L., Scharstein, D.: On learning conditional random fields for stereo: Exploring model structures and approximate inference. IJCV **99**(3), 319–337 (2012)

24. Park, M.G., Yoon, K.J.: Leveraging stereo matching with learning-based confidence measures. In: CVPR, pp. 101–109 (2015)
25. Pfeiffer, D., Gehrig, S., Schneider, N.: Exploiting the power of stereo confidences. In: CVPR, pp. 297–304 (2013)
26. Reynolds, M., Dobos, J., Peel, L., Weyrich, T., Brostow, G.: Capturing time-of-flight data with confidence. In: CVPR, pp. 945–952 (2011)
27. Sabater, N., Almansa, A., Morel, J.: Meaningful matches in stereovision. PAMI **34**(5), 930–942 (2012)
28. Scharstein, D., Pal, C.: Learning conditional random fields for stereo. In: CVPR (2007)
29. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV **47**(1-3), 7–42 (2002)
30. Spyropoulos, A., Komodakis, N., Mordohai, P.: Learning to detect ground control points for improving the accuracy of stereo matching. In: CVPR, pp. 1621–1628 (2014)
31. Sun, X., Mei, X., Jiao, S., Zhou, M., Wang, H.: Stereo matching with reliable disparity propagation. In: 3DIMPVT, pp. 132–139 (2011)
32. Trinh, H., McAllester, D.: Unsupervised learning of stereo vision with monocular depth cues. In: BMVC (2009)
33. Wang, L., Yang, R.: Global stereo matching leveraged by sparse ground control points. In: CVPR, pp. 3033–3040 (2011)
34. Yamaguchi, K., Hazan, T., McAllester, D., Urtasun, R.: Continuous markov random fields for robust stereo estimation. In: ECCV, pp. V: 45–58 (2012)
35. Yoon, K., Kweon, I.: Adaptive support-weight approach for correspondence search. PAMI **28**(4), 650–656 (2006)
36. Zagoruyko, S., Komodakis, N.: Learning to compare image patches via convolutional neural networks. In: CVPR (2015)
37. Zbontar, J., LeCun, Y.: Computing the stereo matching cost with a convolutional neural network. In: CVPR (2015)
38. Zhang, K., Lu, J., Lafruit, G.: Cross-based local stereo matching using orthogonal integral images. IEEE Transactions on Circuits and Systems for Video Technology **19**(7), 1073–1079 (2009)
39. Zhang, L., Seitz, S.: Estimating optimal parameters for mrf stereo from a single image pair. PAMI **29**(2), 331–342 (2007)