Feature Augmentation for Learning Confidence Measure in Stereo Matching

Sunok Kim, *Student Member, IEEE,* Dongbo Min, *Senior Member, IEEE,*
Seungryong Kim, *Student Member, IEEE,* and Kwanghoon Sohn, *Senior Member, IEEE*

***Abstract*—Confidence estimation is essential for refining stereo matching results through a post-processing step. This problem has recently been studied using a learning-based approach, which demonstrates a substantial improvement on conventional simple non-learning based methods. However, the formulation of learning-based methods that individually estimates the confidence of each pixel disregards the spatial coherency that may exist in the confidence map, thus providing a limited performance under challenging conditions. Our key observation is that the confidence features and resulting confidence maps are smoothly varying in the spatial domain and highly correlated within the smooth regions of an image. We present a new approach that imposes spatial consistency on the confidence estimation. Specifically, a set of robust confidence features is extracted from each decomposed superpixel using the Gaussian mixture model (GMM), and then these features are concatenated with pixel-level confidence features. The features are then enhanced through adaptive filtering in the feature domain. In addition, the resulting confidence map, estimated using the confidence features with a random regression forest, is further improved through K-nearest neighbor (K-NN) based aggregation on both the pixel- and superpixel-level. To validate the proposed confidence estimation scheme, we employed cost modulation or ground control points (GCPs) based optimization in stereo matching. Experimental results demonstrate that the proposed method outperforms state-of-the-art approaches on various benchmarks including challenging outdoor scenes.**

***Index Terms*—confidence measure, confidence feature augmentation, confidence map aggregation, ground control point, random regression forest.**

I. Introduction

STEREO matching has long been an important and fundamental topic in the field of computer vision. However, the estimation of accurate corresponding pixels between stereo image pairs in real-world stereo data remains an unsolved problem in particular in the presence of textureless or repeated pattern regions, and occlusions [1]–[3]. Furthermore, photometric variations, such as illumination changes and sensor noise, pose considerably more challenges [4]–[6]. These issues hinder the application to practical systems.

In order to address these issues, numerous approaches [4]–[6], [8]–[11] have been proposed that focus primarily on developing robust matching cost measures. The use of these robust cost measures, however, does not fully address the inherent problem of stereo matching. Although the application of certain popular regularization techniques, such as semi-global matching (SGM) [12] and cost volume filtering (CVF) [13] could facilitate the estimation of a reliable solution, it also faces similar limitations.

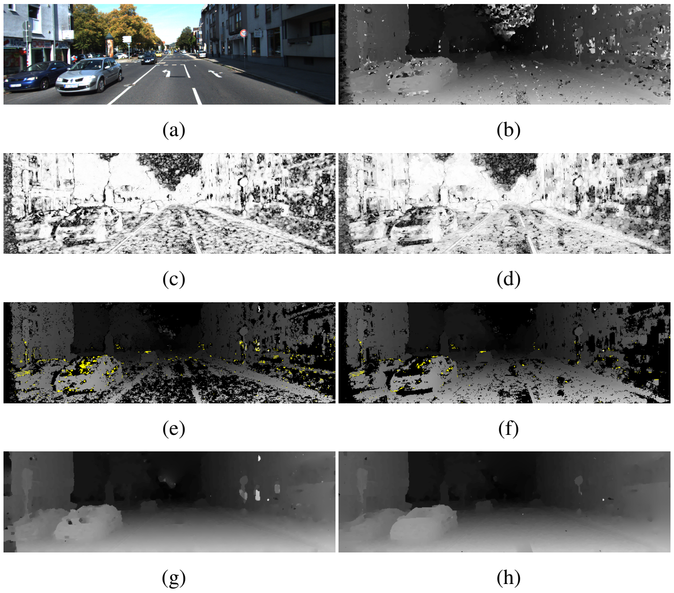


Fig. 1. Importance of a spatial coherency in the confidence estimation: (a) left color image, (b) an initial disparity map estimated using census transform and CVF. Confidence maps were estimated using (c) the per-pixel classifier [7] and (d) the proposed method, respectively. (e) and (f) represent disparity maps consisting of disparity values classified as 'reliable' by thresholding using (c) and (d). Erroneous pixels, which are not determined as inaccurate disparity estimates using the confidence maps, are marked as 'yellow'. (g) and (h) represent disparity maps refined using the confidence maps in (c) and (d), respectively. By leveraging the spatial coherency, the proposed method increases a true negative (TN) accuracy, achieving a significant performance gain on both the confidence estimation and the disparity map refinement.

To alleviate this problem, almost all stereo matching methods involve a post-processing step, where first mismatched pixels are detected using a confidence measure and then these regions are filled with their neighbor information [12]–[16]. Several methods [7], [14]–[19] have been proposed for detecting mismatched pixels by leveraging various confidence features. Among them, learning-based approaches [7], [16], [19], which train a per-pixel classifier with a set of confidence features, demonstrate distinct strengths as compared to existing simple non-learning based approaches [20] (*e.g.*, left-right consistency or peak ratio). In these learning-based approaches, the manner in which a set of informative confidence features is combined for maximizing the performance of the classifier is critical. [16], [19], [21]. In particular, Park and Yoon [7]

S. Kim, S. Kim, and K. Sohn are with the School of Electrical and Electronic Engineering, Yonsei University, Seoul 120-749, Korea (e-mail: kso428@yonsei.ac.kr; srkim89@yonsei.ac.kr; khsohn@yonsei.ac.kr).

D. Min is with the School of Computer Science and Engineering, Chungnam National University, Daejeon 305-764, Korea (e-mail: dbmin@cnu.ac.kr).

extracted a subset of the most important confidence measrues by using the permutation importance accuracy measure [22], and then used these confidence measures as input for training the classifier based on a random decision forest model [23], [24]. However, their method simply makes a decision about the confidence of each pixel independently without considering the spatial coherency that may exist in the confidence map. Thus, such a classifier is still insufficient for reliably detecting mismatched pixels under challenging conditions.

In general, most stereo matching algorithms employ a spatial smoothness assumption to produce a more accurate disparity map [25]. For instance, global methods [12], [26], [27] explicitly incorporate the spatial smoothness assumption in a global manner into an energy-minimization framework. In local methods [13], matching costs are locally smoothed by applying edge-preserving filters, yielding a considerable accuracy gain. In this regard, it is natural to assume that the confidence maps estimated from most stereo matching methods also tend to be spatially smooth, except for object boundary regions. This observation provides useful cues for designing a robust confidence estimator.

In this paper, we propose a new approach for obtaining confidence measures that imposes spatial consistency on the confidence estimation. Our confidence estimator consists of two key ingredients: 1) confidence feature augmentation and 2) confidence map aggregation. Specifically, we impose spatial coherency on the confidence estimation by combining confidence features from both the pixel- and superpixel-level. Gouveia *et al.* [28] also proposed a superpixel-based confidence estimation, but their method involves a superpixel-based disparity fitting process and thus loses pixel-level precision in the confidence estimation. In contrast, our confidence feature is augmented by using the confidence feature computed at the superpixel-level together with the pixel-level confidence feature. Moreover, the superpixel is further decomposed by applying a Gaussian mixture model (GMM) using the pixel-level confidence features to effectively deal with superpixel segmentation errors and outliers caused by occluded pixels within the superpixel. The proposed scheme achieves improved the robustness by imposing spatial coherence while maintaining the pixel-level estimation details. Furthermore, we alleviate the fluctuation of confidence features by applying efficient high-dimensional filtering. In the testing phase, the confidence map is refined using the confidence values of the K-nearest neighbors (K-NN) for each pixel (or superpixel) under the assumption that pixels with similar confidence features are likely to have similar confidence values.

Fig. 1 shows the effectiveness of the proposed confidence estimator. Fig. 1(b) shows an example of a spatially smooth initial disparity map. While the existing per-pixel confidence estimator [7] frequently produces undesired results, in particular for large-textureless regions and occlusions, as shown in Fig. 1(c), the proposed confidence estimator reliably estimates mismatched pixels by leveraging the spatial coherency as shown in Fig. 1(d). The experimental results show that our approach consistently outperforms existing pixel-level confidence measure methods on various benchmarks.

II. Related Works

Numerous approaches have been proposed based on various confidence features [7], [16], [17], [19], [29], [30]. A comparative study of confidence features was provided in [30], where 17 individual confidence features, *e.g.*, left right difference (LRD), matching score measure (MS), and peak ratio (PKR), were analyzed by grouping them into 5 categories and evaluating their capability to predict the mismatched pixel in an estimated disparity map. In [30], it was concluded that there is no single confidence feature that can reliably estimate the confidence map across various scenes.

To alleviate these limitations, methods have been proposed for increasing the accuracy of mismatched pixels prediction in which individual confidence features are combineded. Pfeiffer *et al.* [31] utilize stereo confidence cues from three confidence metrics, peak-ratio naive (PKRN), maximum likelihood metric (MLM), and local curve (LC) information, and improved the accuracy of disparity maps by propagating all the confidence values together with the measured disparities in a Bayesian framework. However, the combination of above three confidence measure is not always effective and thus, it is difficult to apply this algorithm for various databases.

Recently, several approaches [7], [16], [19] that attempt to train a confidence classifier from training data and determine the mismatched pixels using the learned classifier were proposed. In general, these approaches first define the confidence feature vector, which consists of a set of different confidence features, and then train a simple classifier, *e.g.*, random decision forest [23], using the defined confidence features and ground truth confidence values. Haeusler *et al.* [19] combined multiple confidence features and learned the classifier on the correctness of the output disparities. Spyropoulos and Modorhai [16] combined diverse confidence features into a feature vector similarly to the method presented in [19]. They defined confident pixels as ground control points (GCPs) and formulated an a global optimization in a Markov random field (MRF) framework. However, the performance of the method is still limited because of the unreliable confidence features and classifiers. These issues can be addressed by selecting the best set of confidence features among multiple confidence features. Park and Yoon [7] utilized the regression forest framework to select the most important set of confidence features and then trained the regression forest classifiers again to predict the confidence of matching pixels using the selected confidence features. However, all these methods determine the confidence at pixel-level without a spatial constraint, which also limits the detection performance. Gouveia *et al.* [28] proposed an approach for constructing the confidence features by leveraging superpixel-based disparity voting. Although their method extracts meaningful confidence features from superpixels, the results lose pixel-level precision and are very sensitive to segmentation errors. Recently, Seki *et al.* [32] proposed convolutional neural networks (CNNs) based on a confidence estimator. However, they used only the disparity map to predict confidences, and thus, their method was limited in terms of learining optimal confidence features.

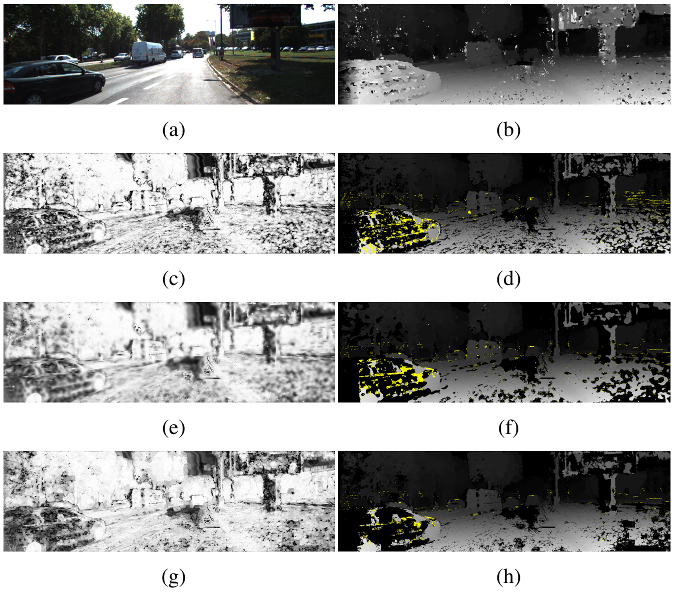


Fig. 2. Visual comparison of confidence maps: (a) left color image, (b) an initial disparity map. (c) represents a confidence map estimated using the per-pixel classifier [7]. (e) shows a confidence map refined by simply applying the guided filter [33]. (g) represents our result. (d), (f), and (h) represent disparity maps consisting of disparity values classified as 'reliable' by thresholding using (c), (e), and (g) respectively. Erroneous pixels, which are not determined as inaccurate disparity estimates using the confidence maps, are marked as 'yellow'. The AUC of (c), (e), and (g) is 0.049, 0.046, and 0.041 respectively. The optimal AUC with a ground truth confidence map is 0.025.

III. Proposed Method

A. Problem Formulation and Overview

Let us define stereo image pairs $I_i^L, I_i^R : \mathcal{I} \rightarrow \mathbb{R}^3$ for pixel $i = [i_\mathrm{x}, i_\mathrm{y}]^T$, where $\mathcal{I} \subset \mathbb{N}^2$ is a discrete image domain. Using existing stereo matching approaches, we first compute a 3-D cost volume $C_{i,d}$ with a disparity $d \in \{1, ..., d_\mathrm{max}\}$, where d_max is a maximum allowed disparity, and estimate its associated disparity map $D_i = \mathrm{argmin}_d C_{i,d}$. A ground truth confidence map Q_i^* is then computed by simply comparing the estimated disparity map D_i and the ground truth disparity map D_i^*. The objective of this study was to estimate a reliable confidence $Q_i \in [0, 1]$, which represents the confidence of D_i. Formally, it can be estimated by learning the classifier $\mathcal{R} : f_i \rightarrow Q_i^*$ with a confidence feature $f_i : \mathcal{I} \rightarrow \mathbb{R}^{N_f}$ and the ground truth confidence map Q_i^*. The confidence feature f_i is typically defined using the cost volume $C_{i,d}$ and/or its corresponding disparity D_i, where N_f is the dimension of the confidence feature.

Most existing stereo matching methods construct the cost volume by employing the spatial smoothness constraint both globlayy and locally [25]. In global methods [12], they apply an explicit smoothness assumption and find an optimal disparity by solving an MRF-based optimization problem. Local methods [10], [13] compute the disparity within a finite support region by aggregating the pixel-based matching cost to implicitly impose the smoothness assumption. In this context, the confidence map estimated from the smoothly varying disparity map also tends to exhibit similar attributes, as exemplified in Fig. 1. However, conventional approaches [7], [16], [19] estimate the confidence map by utilizing a confidence feature vector defined on each pixel independently without considering any spatial dependency, thus frequently producing poor results. A structured learning framework has been used to provide structured (*i.e.*, spatially smooth) outputs by taking into account input data from adjacent pixels in the classifier [34]–[36]. For instance, in [34], a spatially coherent edge map is estimated by designing the structured classifier that takes advantage of the structure present in local image patches. In our work, confidence features often contain severe, dense outliers because of occlusion, which poses challenges for designing a reliable structured classifier. Instead of designing a structured classifier for the confidence estimation, in this study we aimed at improving the performance of the confidence classification even with a conventional classifier (*e.g.*, random regression forest) by exploiting spatially coherent confidence features.

The design of such confidence features is, however, a non-trivial task. One simple solution is to apply existing edge-preserving filters (*e.g.*, guided filter [33] or bilateral filter [37]) for aggregating the pixel-level confidence feature (or confidence map) in a locally adaptive manner, as in cost aggregation schemes in stereo matching. However, this simple aggregation is not very effective because of the nonlinearity of confidence features and dense outliers incurred by occlusion. Fig. 2 shows that such simple filtering is not effective. Although we assume that the confidence map is spatially smooth, a conventional smoothing filter cannot improve the performance of the confidence measure, as shown in Fig. 2(f). However, the proposed method effectively removes mismatched pixels as shown in Fig. 2(h).

In our approach, we first decompose a color image using an off-the-shelf superpixel decomposition method for generating superpixel-level confidence features under the assumption that pixels belonging to the same superpixel are likely to have similar confidence features and confidence values. To effectively address superpixel segmentation errors (due to hard decisions on object boundaries) and dense outliers such as occluded pixels within a superpixel, we generate a set of reliable confidence features through clustering based on the GMM within each superpixel. For training the confidence classifier, we adaptively combine the pixel-level and superpixel-level confidence features for each pixel. Such adaptive confidence features considerably increase the robustness of the classifier, while retaining the pixel-level precision on the estimated confidence map.

These robust confidence features are also beneficial in the testing phase. The estimated confidence map is further enhanced by leveraging the correlation between the confidence features and the confidence map. Namely, pixels with similar confidence features are highly like to have similar confidence values. We thus refine the estimated confidence value through an adaptive summation weighted by K-NN confidence features in a multi-scale manner. The overall framework of our approach is summarized in Fig. 3.

B. Confidence Feature Augmentation (CFA)

To design a more distinctive confidence feature, we use the confidence features computed at both the pixel- and

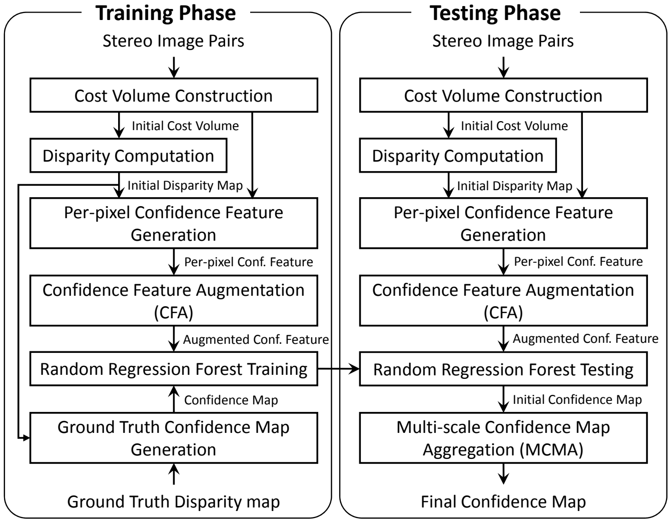


Fig. 3. Framework of the proposed confidence estimation method, consisting of confidence feature augmentation (CFA) and multi-scale confidence map aggregation (MCMA).

superpixel-level. A confidence classifier is then simply trained using a random regression forest [24].

1) Pixel-level Confidence Feature: We start with the pixel-level confidence feature vector f_i for a pixel i. We select the set of the most important confidence features from various confidence feature candidates as in [7]. From an estimated cost volume $C_{i,d}$ derived with a cost function (*e.g.*, census transform [4]) and its corresponding disparity map D_i, we construct a confidence feature vector f_i with the ensemble of independent confidence features. We use 8×1 confidence feature vectors used in [7], which utilize two different feature sets according to the database. 'feature selection 1' is composed of LRC, the distance to the border, LRD, the median disparity deviation values (MDDs) in three different scales, MLM, and the MS, and 'feature selection 2' is composed of MDDs in four different scales, LRD, MLM, PKRN, and the negative entropy measure. It should be noted that our framework can be incorporated with any other confidence features.

2) Superpixel-level Confidence Feature: As mentioned earlier, we consider a spatial smoothness assumption for confidence features. To this end, we propose two stage image decomposition strategy. We first decompose the reference image, *e.g.*, I^L, into a set of non-overlapping superpixels $\mathbf{S} = \{\mathcal{S}_m | \bigcup_m \mathcal{S}_m = \mathcal{I}, m = 1, ..., M_\mathbf{S}\}$, where $M_\mathbf{S}$ is the number of superpixels. In this study, we used the SLIC superpixel algorithm [38], but any other off-the-shelf superpixel segmentation approaches can be used. Within each superpixel region $\mathcal{S}_m$, we generate a set of reliable confidence features from the pixel-level confidence feature f_i. In order to deal with superpixel segmentation errors and outliers caused by occluded pixels within the superpixel, we further decompose each superpixel into M sub-clusters using the GMM model [39]. For the superpixel $\mathcal{S}_m$, input features for clustering based on the GMM model are formed with the pixel-level confidence feature and spatial information such that $x_i = [f_i^T, i_\mathrm{x}, i_\mathrm{y}]^T$ for $i \in \mathcal{S}_m$. Note that the spatial information $(i_\mathrm{x}, i_\mathrm{y})$ is used together so that spatially adjacent confidence features are grouped.

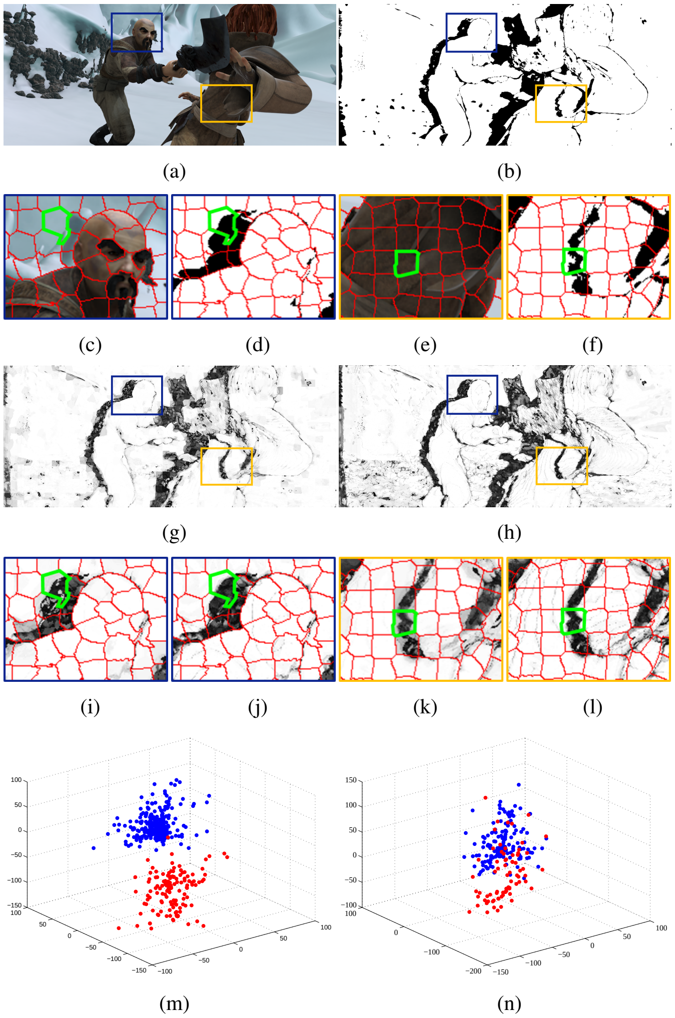


Fig. 4. Effectiveness of the proposed confidence feature augmentation scheme: (a) left color image, (b) ground truth confidence map. (c), (d), (e), and (f) show enlarged windows for (a) and (b) with superpixels overlaid. The superpixels marked with the 'green' contour in (c) and (e) contain occlusion and inaccurate segmentation boundary, respectively. (g) and (h) represent confidence maps estimated using simple SLIC superpixel confidence features ($L = 1$) and confidence features augmented using both SLIC superpixel and GMM clustering ($L = 2$), respectively. (i) and (k) show regions enlarged from the confidence map in (g). (j) and (l) show regions enlarged from the confidence map in (h). (m) and (n) represent the set of 3-dim feature vectors computed in the superpixels with the 'green' boundaries in (c) and (e), respectively. For the purpose of visualization, the 10-dim feature vectors x_i are projected into 3-dim feature vectors. It validates the effectiveness of our approach that re-decomposes the SLIC superpixels through GMM clustering using the set of pixel-level confidence features.

Let us define the GMM parameter $\theta_m = \{\phi^l_m, \mu^l_m, \Sigma^l_m | l = 1, 2, ..., L\}$ of each superpixel $\mathcal{S}_m$. We also define the multinomial Gaussian distribution of x_i as $\mathcal{N}(x_i|\mu^l_m, \Sigma^l_m)$. Then, the likelihood function to x_i with respect to θ_m can be defined such that

$$\Pr(x_i|\theta_m) = \sum\nolimits_l \phi^l_m \mathcal{N}(x_i|\mu^l_m, \Sigma^l_m). \quad (1)$$

Based on the estimated parameters θ_m, we can decompose each superpixel into L sub-clusters. The GMM is built using the expectation-maximization (EM) algorithm [40]. With a sub-cluster label z_l, the sub-cluster probability for each pixel

i within superpixel $\mathcal{S}_m$ can be estimated:

$$\Pr(z_l = 1|x_i) = \frac{\phi_m^l \mathcal{N}(x_i|\mu_m^l, \Sigma_m^l)}{\sum_k \phi_m^k \mathcal{N}(x_i|\mu_m^k, \Sigma_m^k)}, \tag{2}$$

where $k \in \{1, 2, ..., M\}$. With $\Pr(z_l = 1|x_i)$, we determine which sub-cluster the pixel $i \in \mathcal{S}_m$ belongs to by computing $l_i = \text{argmax}_l \Pr(z_l = 1|x_i)$. Then, for the simplicity of notation, we re-define a set of the final superpixels $\mathbf{S}' = \{\mathcal{S}'_n|n = 1, ..., N_{\mathbf{S}'}\}$ and a set of its associated superpixel-level features $\{\mu_n|n = 1, ..., N_{\mathbf{S}'}\}$ over an entire image, where the number of the final superpixels $N_{\mathbf{S}'} = LM_{\mathbf{S}}$. Note that it is assumed that the number of sub-clusters L is fixed for all superpixels.

When the pixel i belongs to $\mathcal{S}'_n$, *i.e.*, $i \in \mathcal{S}'_n$, the augmented confidence feature g_i is built by concatenating the pixel-level confidence feature f_i and the superpixel-level feature μ_n as follows:

$$g_i = [f_i^T, \mu_n]^T. \tag{3}$$

Note that pixels belonging to the same superpixel have the same superpixel-level confidence feature μ_n. The augmented confidence feature μ_n is more robust to outliers compared to f_i, but it might lose a pixel-level precision on confidence estimation. By simultaneously leveraging the pixel-level confidence feature f_i and superpixel-level confidence feature μ_n, we are capable of encoding the distinctive confidence feature with both pixel- and superpixel-level constraints. The proposed confidence feature preserves the pixel-level precision on the confidence estimation while maintaining the robustness to outliers. Interestingly, a superpixel-based approach has been developed in the stereo matching literature such that disparity fitting results within color segments are additionally used for defining an objective function for global optimization [41], demonstrating highly accurate disparity estimation performance.

To further alleviate the fluctuation of the augmented confidence feature g_i, we filter it out, where the filter kernel is based on weights defined from confidence feature itself. Here, we apply adaptive manifold filter to efficiently perform high-dimensional filtering [42] such that

$$\widetilde{g}_i = \sum\nolimits_{j \in \mathcal{N}_i} W_{i,j}^g g_j, \tag{4}$$

where $W_{i,j}^g$ is the weight function representing the confidence feature similarity, defined such that

$$W_{i,j}^g = \frac{\exp(-\|g_i - g_j\|^2/\sigma_g)}{\sum_{j \in \mathcal{N}_i} \exp(-\|g_i - g_j\|^2/\sigma_g)}, \tag{5}$$

where $\mathcal{N}_i$ is the local window of filter and σ_g is the standard deviation of the Gaussian function. Note that we define the weight $W_{i,j}^g$ using only confidence feature without considering spatial distances, since we use the confidence features within the local window $\mathcal{N}_i$ in defining weights.

In Fig. 4, we demonstrate the effectiveness of the proposed confidence feature augmentation scheme. Two cases that may frequently occur in the superpixels are considered: occlusion, as shown in Fig. 4(c), and segmentation errors, as shown in Fig. 4(e). Simply generating superpixel confidence features within these superpixels, *i.e.*, $L = 1$, may lead to undesired artifacts in the confidence estimation as shown in Fig. 4(g). Such artifacts were successfully removed in the confidence map of Fig. 4(h), estimated using confidence features augmented by using both the SLIC superpixel and GMM clustering ($L = 2$). We analyzed this in more depth using pixel-level confidence features within SLIC superpixels. For the purpose of visualization, we first apply the principal component analysis (PCA) [43] to $x_i = [f_i^T, i_\text{x}, i_\text{y}]^T$ for $i \in \mathcal{S}_m$, which is a 10-dimensional feature vector, and then visualize them with three dominant PCA coefficients only. We can clearly see that the pixel-level confidence features are separated into two distinct classes through the GMM clustering. This validates the effectiveness of our approach that re-decomposes the SLIC superpixels through GMM clustering using the set of pixel-level confidence features. We found setting $L = 2$ yields the best performance in the confidence estimation. More detailed analysis will be given in experiments.

3) Confidence Classifier Learning: Finally, using the robust augmented confidence feature $\widetilde{g}_i$ and ground truth confidence map Q_i^*, we train the random regression forest $\mathcal{R}$ satisfying

$$\mathcal{R} : \widetilde{g}_i \to Q_i^*. \tag{6}$$

We will show that the proposed confidence feature outperforms existing approaches.

C. Multi-Scale Confidence Map Aggregation (MCMA)

In order to further improve the quality of the confidence map, we utilize the proposed confidence feature once again in the testing phase by exploiting a multi-scale confidence map aggregation scheme at both the pixel- and superpixel-level. As in confidence feature generation described in Sec. III-B, we leverage the correlation between the confidence features and confidence values by assuming that pixels with similar confidence features are likely to have similar confidence values.

In a symmetric viewpoint of the confidence feature augmentation, we propose an aggregation scheme on an estimated confidence map Q_i that reuses the weight of $\widetilde{g}_i$ at both the pixel- and superpixel-level. We first define the superpixel-level confidence feature $\widetilde{g}_n$ and confidence value Q_n by averaging within each superpixel $\mathcal{S}'_n$ such that $\widetilde{g}_n = \sum_{i \in \mathcal{S}'_n} \widetilde{g}_i / |\mathcal{S}'_n|$ and $Q_n = \sum_{i \in \mathcal{S}'_n} Q_i / |\mathcal{S}'_n|$, where $|\mathcal{S}'_n|$ is the number of pixels within $\mathcal{S}'_n$. In order to apply multi-scale aggregation, we first define augmented confidence feature set g^A and augmented confidence value set Q^A as follows:

$$g^\text{A} = \{\widetilde{g}_i \cup \widetilde{g}_n | i \in \mathcal{I}, n \in 1, ..., N_{\mathbf{S}'}\}, \tag{7}$$

$$Q^\text{A} = \{Q_i \cup Q_n | i \in \mathcal{I}, n \in 1, ..., N_{\mathbf{S}'}\}. \tag{8}$$

K-NN nearest neighbors for all pixels and superpixels are estimated using the K-nearest neighbor search [44]. The confidence value is filtered by averaging the confidence values of K-NN neighborhoods:

$$\widetilde{Q}_u^\text{A} = \sum\nolimits_{v \in \mathcal{N}_u} Q_v^\text{A} / K, \tag{9}$$

where u represents an index for all pixels and superpixels within an image, and $\mathcal{N}_u$ is K-NN nearest neighborhoods for u.

```
Algorithm 1: Feature Augmentation for Learning Confidence Measure
Input: training set $\mathcal{C}_{\text{train}} = \{I_i^L, I_i^R, D_i^*\}$, testing pairs $\{I_i^L, I_i^R\}$
Output: confidence map $\widehat{Q}_i$, disparity map $\widehat{D}_i$

    /* Training Procedure */
1 : Construct cost volume $C_{i,d}$ and disparity map $D_i$ for $\mathcal{C}_{\text{train}}$.
2 : Calculate ground truth confidence map $Q_i^*$ by thresholding
    the difference between $D_i$ and $D_i^*$.
3 : Construct the augmented confidence feature $g_i$ by pixel-level
    confidence feature $f_i$ and superpixel-level confidence feature $\mu_n$.
4 : Filter out the confidence feature $g_i$ to generate $\widetilde{g}_i$ in Eq. (4).
5 : Train the random regression forest $\mathcal{R}$ using $\widetilde{g}_i$ and $Q_i^*$ in Eq. (6).

    /* Testing Procedure */
6 : Estimate $Q_i$ using confidence features $\widetilde{g}_i$ and $Q_i^*$ for testing stereo
    pairs $\{I_i^L, I_i^R\}$ through step 1-5.
7 : Compute $\widetilde{Q}_i$ and $\widetilde{Q}_n$ using multi-scale confidence map aggregation
    on $Q_i$ and $Q_n$ in Eq. (9).
8 : Compute $\widehat{Q}_i$ with the weighted sum of $\widetilde{Q}_i$ and $\widetilde{Q}_n$ in Eq. (10).
9 : Compute refined disparity map $\widehat{D}_i$ using cost modulation based
    optimization or GCPs-based optimization.
```

Our final confidence value $\widehat{Q}_i$ for each pixel i is then the weighted sum of $\widetilde{Q}_i$ and $\widetilde{Q}_n$ as follows:

$$\widehat{Q}_i = \alpha \widetilde{Q}_i + (1-\alpha)\widetilde{Q}_n, \quad i \in \mathcal{S}_n', \tag{10}$$

where α is the weight parameter between the pixel-level confidence value and superpixel-level confidence value. Algorithm 1 summarizes the proposed confidence measure, consisting of confidence feature augmentation (CFA) and multi-scale confidence map aggregation (MCMA).

IV. Validation

So far, we have explained the method for obtaining a reliable confidence measure from the estimated initial disparity map. In this section, we describe the validation of the effectiveness of our confidence measure, for which we incorporated the estimated confidence values into optimization schemes, by using 1) cost modulation based optimization as in [7] and 2) GCPs-based global optimization as in [16], [31].

A. Cost Modulation Based Optimization

Borrowed from [7], we incorporate the predicted confidence value into stereo matching by modulating the initial matching costs. When $C_{i,d}$ denotes the matching cost of pixel i for a disparity d, it is first modulated using $\widehat{Q}_i$ such that

$$\widehat{C}_{i,d} = \widehat{Q}_i C_{i,d} + (1-\widehat{Q}_i)\sum\nolimits_d C_{i,d}/d_{\max}. \tag{11}$$

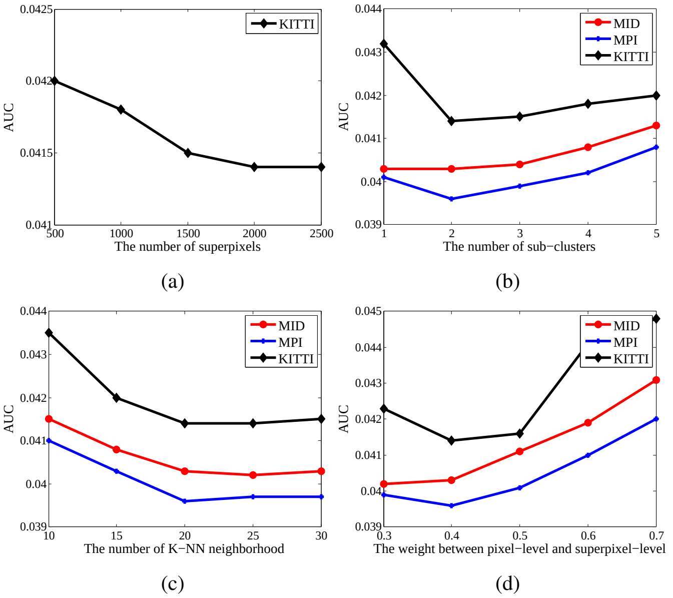
After modulating the initial cost volumes, the matching costs of confident pixels remain unchanged and unreliable pixels are flattened. Therefore, unreliable pixels can be easily dominated by more confident neighboring pixels in the optimization step. To produce a final disparity map $\widehat{D}_i$, the modulated cost function $\widehat{C}_{i,d}$ is then optimized using a global approaches such as SGM [12] or belief propagation [45].

B. GCPs-Based Optimization

Our confidence measure can also be incorporated in GCPs-based optimization. We first set pixels that have a higher value than the threshold δ as the GCPs and then globally optimize the initial GCPs through an MRF-based propagation. As mentioned in [16], there is a trade-off between density and accuracy. Namely, with a low threshold, the density of GCPs becomes high but the true negative (TN) accuracy is degraded and vice versa. To prevent disparity errors from being propagated in the optimization, we focus on increasing the TN accuracy by using a relatively high threshold, although this degrades the true positive (TP) accuracy. With the set of GCPs, we define the energy function as in [50] to obtain the final disparity map as follows:

TABLE I
Experimental Configuration.

Dataset	MID [46]	MPI [47]	KITTI [48]	ZED
Confidence Feature	Feature Selection 1		Feature Selection 2	
Training Set	MID 2005		KITTI 2012 (8 frame)	
Testing Set	MID 2006 21 images	MPI 22 frames	KITTI 2015 200 frames	ZED stereo



(a) (b)

(c) (d)

Fig. 5. Average AUC of proposed confidence measure on MID [46], MPI [47], and KITTI [49] benchmark, as varying (a) the number of superpixels, (b) the number of sub-clusters M, (c) the number of K-NN nearest neighborhoods K, and (d) the pixel-level weight α.

$$E(\widehat{\mathbf{D}}) = (\widehat{\mathbf{D}} - \mathbf{W}\mathbf{D})^T(\widehat{\mathbf{D}} - \mathbf{W}\mathbf{D}) + \lambda\widehat{\mathbf{D}}\mathbf{L}\mathbf{D}, \tag{12}$$

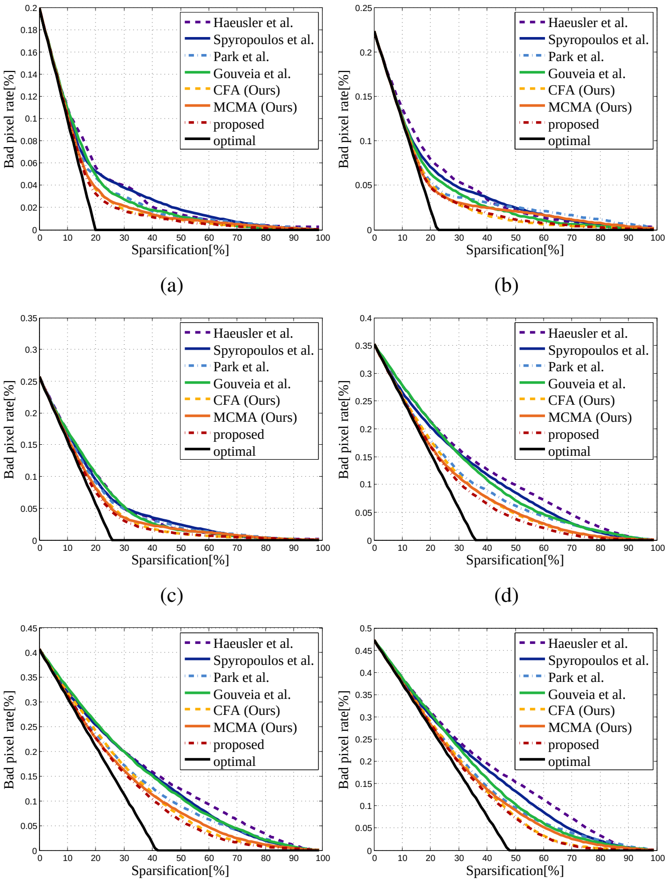
where $\mathbf{D}$ and $\widehat{\mathbf{D}}$ is the vector form of the estimated initial disparity map D_i and output disparity map $\widehat{D}_i$, respectively. The weight matrix $\mathbf{W}$ is composed with the component $\mathbf{W}_{i,j}$ between pixel i and j is defined as

$$\mathbf{W}_{i,j} = \frac{m_i k_{i,j} \exp(-\|D_i - D_j\|^2/\sigma_D)}{\sum_{j \in \mathcal{N}_i^4} m_i k_{i,j} \exp(-\|D_i - D_j\|^2/\sigma_D)}, \tag{13}$$

where m_i is a binary mask to mark the position of GCPs, *i.e.*, it is 1 for GCPs and 0 otherwise, and $k_{i,j}$ is the affinity between the pixel i and j in the feature space of color intensity and spatial location. σ_D is the standard deviation of the Gaussian function, and $\mathcal{N}_i^4$ represents a local 4-neighborhood. $\mathbf{L}$ is the sparse Laplacian for regularization where each element $\mathbf{L}_{i,j} = -k_{i,j}$ for $i \neq j$ and $\mathbf{L}_{i,i} = \sum_{j \in \mathcal{N}_i^4} k_{i,j}$. With this

TABLE II
THE AVERAGE AUC VALUES FOR MID [46], MPI [47], AND KITTI [49] DATASET. THE AUC VALUE OF GROUND TRUTH CONFIDENCE IS MEASURED AS 'OPTIMAL'. THE RESULT WITH THE LOWEST AUC VALUE IN EACH EXPERIMENT IS HIGHLIGHTED.

Methods	MID [46]		MPI [47]		KITTI [49]		
	Census		Census		Census		MC-CNN
	w/SGM	w/CVF	w/SGM	w/CVF	w/SGM	w/CVF	w/CBCA
Haeusler *et al.* [19]	0.0530	0.0512	0.0532	0.0499	0.0556	0.0460	0.0124
Spyropoulos *et al.* [16]	0.0485	0.0468	0.0482	0.0454	0.0503	0.0421	0.0124
Park *et al.* [7]	0.0456	0.0439	0.0450	0.0424	0.0469	0.0348	0.0108
Gouveia *et al.* [28]	0.0706	0.0684	0.0700	0.0672	0.0748	0.0451	-
CFA (Ours)	0.0416	0.0399	0.0409	0.0387	0.0429	0.0288	0.0097
MCMA (Ours)	0.0433	0.0416	0.0427	0.0402	0.0446	0.0302	0.0100
CFA+MCMA (Ours)	**0.0403**	**0.0386**	**0.0396**	**0.0374**	**0.0414**	**0.0269**	**0.0096**
Optimal	0.0340	0.0322	0.0335	0.0312	0.0348	0.0166	0.0038



(a) (b)

(c) (d)

(e) (f)

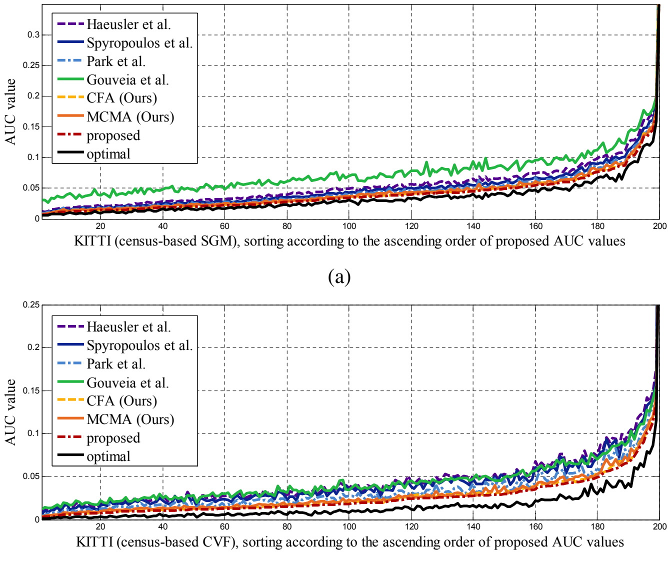
Fig. 6. Sparsification curve for (a) Bowling2, (b) Flowerpots, and (c) Lampshade1 are selected images from MID [46], (d) frame 6, (e) frame 17, and (f) frame 155 are selected images from KITTI dataset [49]. The sparsification curve for a ground truth confidence map is described as 'optimal'.

simple quadratic optimization scheme, we obtain a highly reliable disparity map.

V. EXPERIMENTAL RESULTS

A. Experimental Settings

In this section, we compare the proposed method with conventional learning-based approaches [7], [16], [19], [28] on various dataset [46], [47], [49]: Middlebury 2006 (MID), MPI-Sintel (MPI), KITTI 2015 (KITTI), and ZED, which is described below. We also apresent an anlaysis of the performance gain resulting from the two key components of our method (CFA and MCMA). To compute a raw matching cost, we used a census transform with a 5×5-sized local window and the matching cost with a convolutional neural network (MC-CNN) method [11]. 'KITTI 2015 fast network' provided by the author's website [51] was used for computing the MC-CNN. For the census transform, we applied SGM [12] or CVF [13] on the estimated cost volume. For the SGM, we set $P_1 = 0.008$ and $P_2 = 0.126$ as in [7]. For the CVF, we performed the cost volume filtering using the guided filter [33] of a 19×19-sized local window and regularization parameter $\epsilon = 0.01^2$ as recommended in [13]. To produce the MC-CNN results, following the original MC-CNN paper in [11], we performed the cost aggregation using an adaptive cross window (CBCA) [52]. Note that the MC-CNN is a top-ranked method in the KITTI benchmark [48], and we demonstrate that the proposed confidence measure can be used to improve the quality of such a top-performing stereo matching algorithm through a post-processing step. For SLIC superpixels, we decompose the image into a different number of superpixels for each dataset: 500 for MID and 2000 for MPI, KITTI, and ZED. In MCMA, we set $K = 20$. For GCPs-based optimization, we set $\sigma_D = 10$ and $\delta = 0.7$ using cross-validation for which we divided the



(a)

(b)

Fig. 7. AUC values of (a) census-based SGM and (b) census-based CVF for KITTI dataset [49]. We sort the AUC values in the ascending order according to proposed AUC values. The 'optimal' AUC values are calculated using ground truth confidence map.

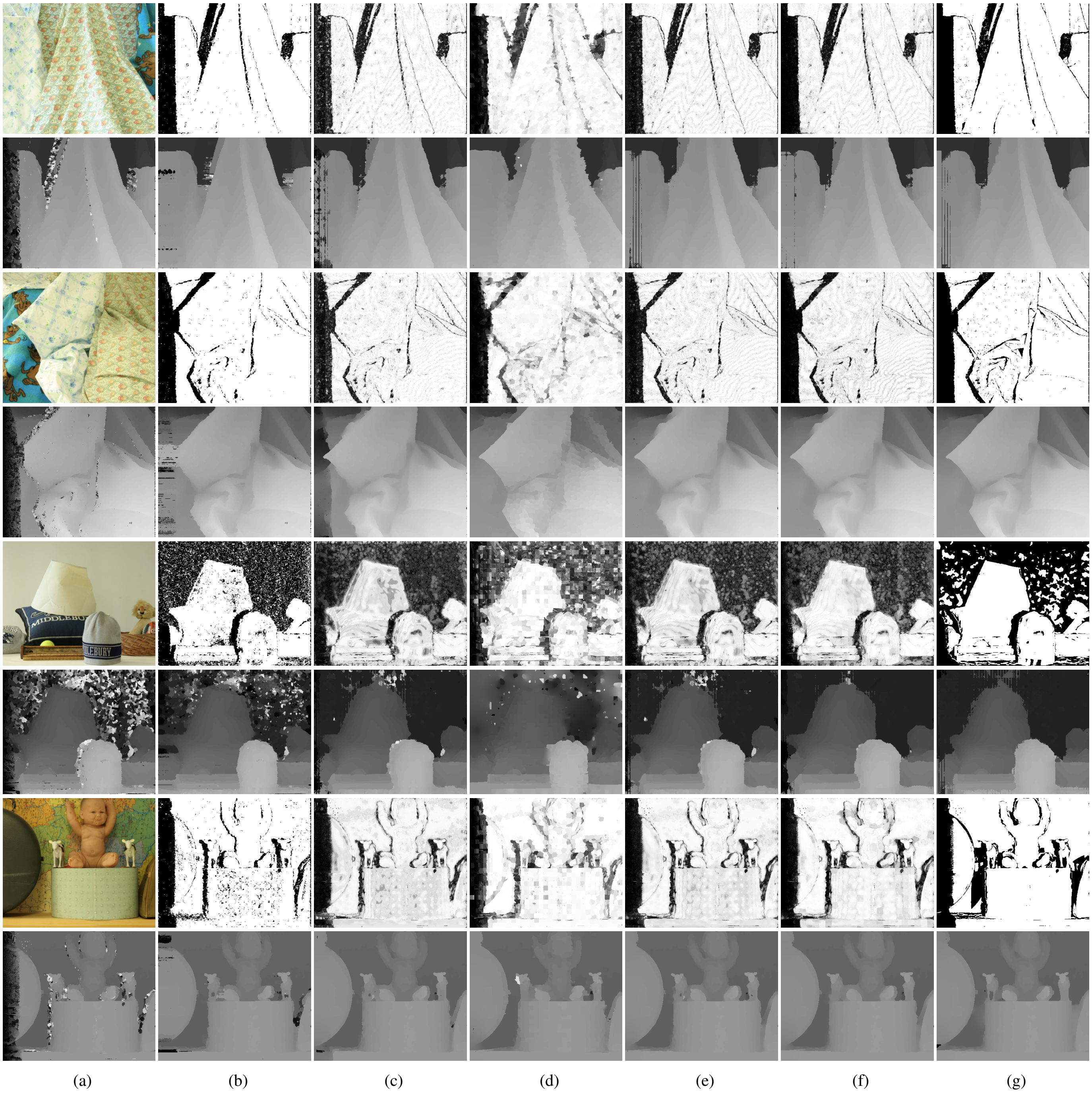


Fig. 8. The confidence maps and refined disparity maps of Middlebury dataset [46] using (from top to bottom) census-based SGM+mod., SGM+GCPs, CVF+mod., and CVF+GCPs. (a) color images and initial disparity maps, refined disparity maps with confidence maps estimated by (b) conventional post-processing (LRC+WMF), (c) Park *et al.* [7], (d) Gouveia *et al.* [28], (e) CFA (Ours), (f) CFA+MCMA (Ours), and (g) ground truth confidence map.

training and testing dataset into three groups, performed three-fold cross-validations, and then estimated the parameters that provide the highest performance. Random forests were trained comprising 50 trees in regression mode, using the Matlab TreeBagger package [53].

For evaluation, we divided the dataset into the MID [46] and MPI [47] dataset taken (synthesized) under carefully-controlled environments, and real-world dataset as KITTI [49] and ZED stereo dataset that we captured. ZED stereo dataset was built with ZED stereo camera [54], where the resolution of the stereo image pairs is full HD (1920×1080). We captured the dataset for outdoor environment, such as a playground, road, and building, etc. For MID and MPI, we trained the classifier using 6 images in the Middlebury 2005 dataset [46] and for KITTI and ZED stereo dataset, we trained the classifier using 8 frames in KITTI 2012 dataset [49] as in [7]. The experimental configuration is summarized in Table I.

To evaluate the performance of the confidence measures quantitatively, we used the sparsification curve and its area under curve (AUC), as in [7], [16], [19]. The sparsification curve draws the bad pixel rates while successively removing the pixels in descending order of confidence measure values in

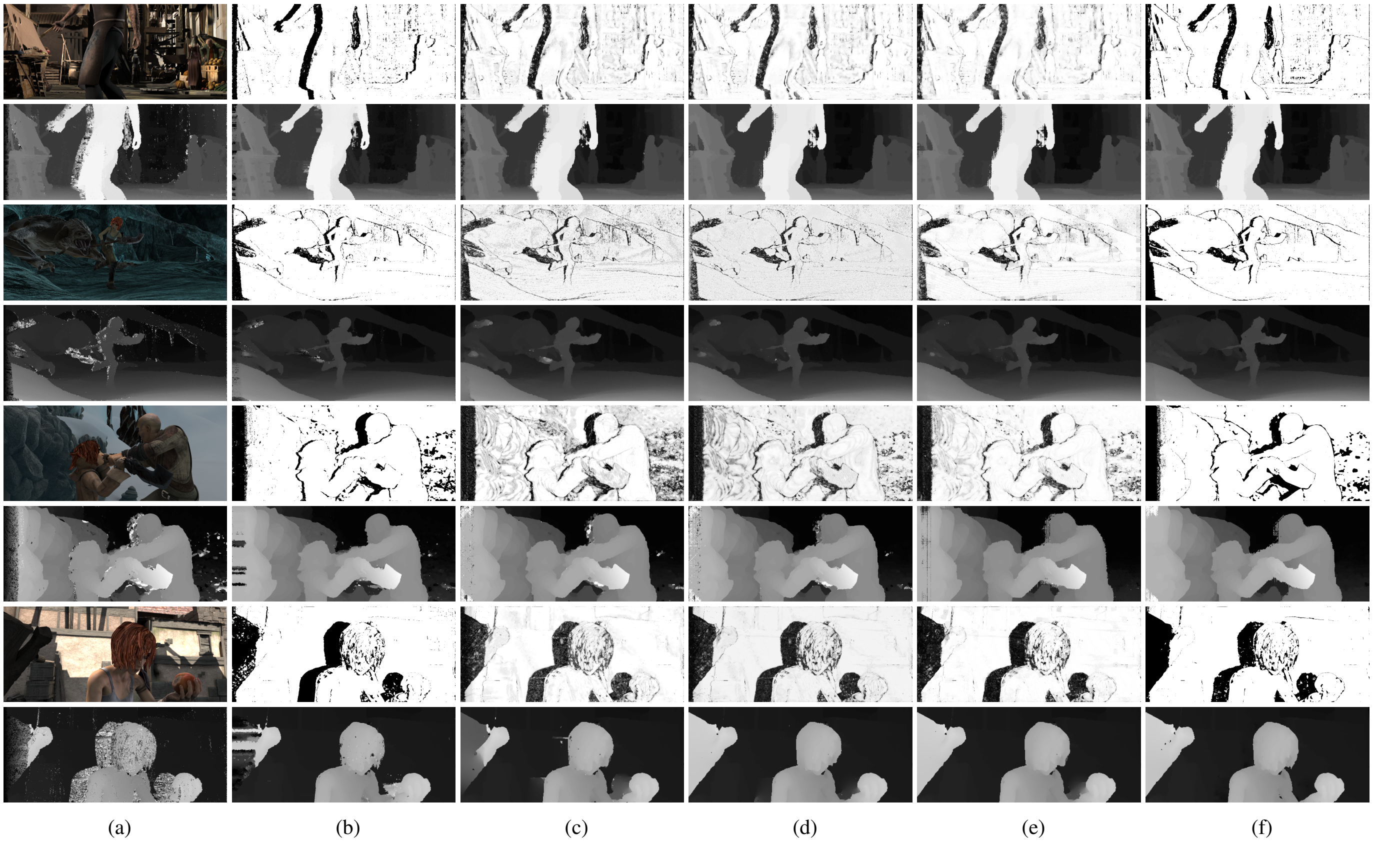


Fig. 9. The confidence maps and refined disparity maps of MPI dataset [47] using (from top to bottom) census-based SGM+mod., SGM+GCPs, CVF+mod., and CVF+GCPs. (a) color images and initial disparity maps, refined disparity maps with confidence maps estimated by (b) Park *et al.* [7], (c) Gouveia *et al.* [28], (d) CFA (Ours), (e) CFA+MCMA (Ours), and (g) ground truth confidence map.

the disparity map, thus it is possible to observe the tendency of prediction errors. AUC quantifies the ability of a confidence measure to estimate correct matches. The higher the accuracy of the confidence measure, the lower the AUC value. To evaluate the quantitative performance, we also measured the average bad matching percentage (BMP) as in [46], [47].

B. Parameter Sensitivity Analysis

We analyzed the performance of the proposed confidence measure while varying the associated parameters, including the number of superpixels $M_{\mathbf{S}}$, the number of sub-clusters L in CFA, the number of K-NN nearest neighborhood K, and the parameter α weighting between the pixel- and superpixel-level confidence maps in MCMA.

We evaluated the average AUC for different numbers of segments in superpixels as shown in Fig. 5(a). If the number of superpixels is too small, diferent regions may be mixed in a single superpixel, producing a low AUC. We segmented an input image into a set of superpixels that contained about 200 pixels on average. The average AUC values for varying L are shown in Fig. 5(b). If L is 1, we generated a single superpixel-level confidence feature in $\mathcal{S}_m$, which may contain inaccurate segmentation errors and occlusion outliers. We found that setting $L = 2$ achieves the best results. As the value of L increases, the AUC value is degraded because of over-fragmented clusters. The average AUC values for varying value of K are shown in Fig. 5(c). The AUC value decreases as the values of K increases and converges when K is 20. We set K as 20 in all experiments. The average AUC value for varying values of α are shown in Fig. 5(d). As the value of α increases, there is a possibility of missing the spatial coherency. We set α as 0.4 in all the experiments.

C. Confidence Measure Analysis

We compared the AUC with conventional learning-based approaches [7], [16], [19], [28]. The optimal AUC can be obtained with a ground truth confidence map. Sparsification curves for selected frames in the MID and KITTI datasets are shown in Fig. 6. The results show that the proposed confidence estimator exhibits a better performance than conventional per-pixel classifiers [7], [16], [19] and a superpixel-based classifier [28]. We used census-based SGM [4], [12] and census-based CVF [4], [13] to obtain the initial disparity maps. The average AUC value with census-based SGM and census-based CVF for MID, MPI, and KITTI is summarized in Table II. The table shows that in all cases the confidence maps estimated with proposed method (CFA + MCMA) have the lowest AUC values, as compared to the per-pixel classifiers [7], [16], [19]. Fig. 7 describes the AUC values, which are sorted in ascending order, for the KITTI dataset. These results demonstrate that the proposed approach outperforms other methods for predicting mismatched pixels. When either CFA or MCMA are applied, our method outperforms conventional confidence measure methods [7], [16], [19], [28], and the

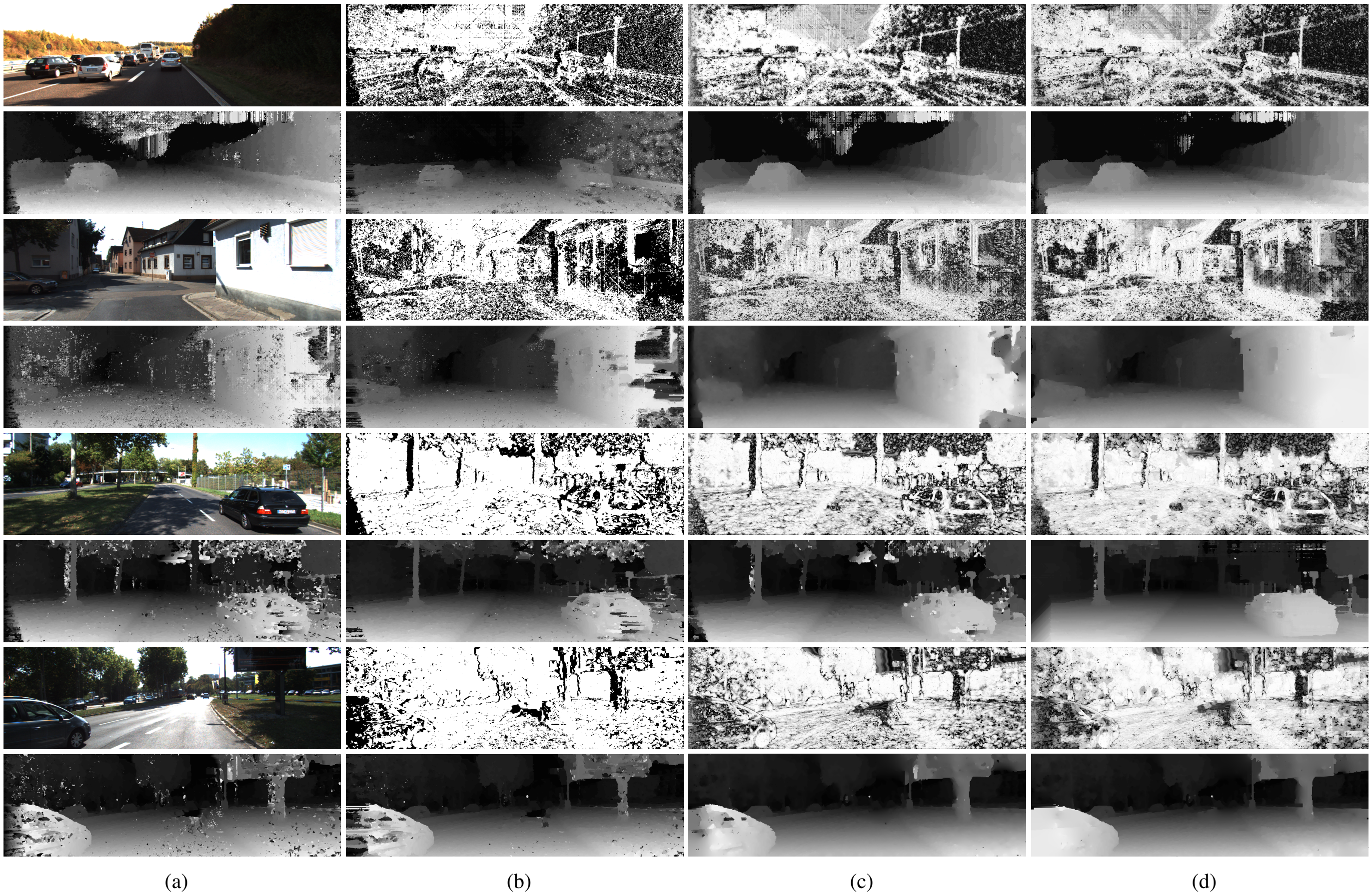


Fig. 10. The confidence maps and refined disparity maps of KITTI dataset [49] using (from top to bottom) census-based SGM+mod., SGM+GCPs, CVF+mod., and CVF+GCPs. (a) color images and initial disparity maps, refined disparity maps with confidence maps estimated by (b) conventional post-processing (LRC+WMF), (c) Park *et al.* [7], and (d) CFA+MCMA (Ours).

TABLE III
THE BMP OF THE RESULTANT DISPARITY MAP FOR MIDDLEBURY [46], MPI [47], AND KITTI [49] DATASET. THE BAD PIXEL ERROR RATE OF THE REFINED DISPARITY MAP USING GROUND TRUTH CONFIDENCE IS MEASURED AS 'OPTIMAL'. THE RESULT WITH THE LOWEST BAD PIXEL ERROR IN EACH EXPERIMENT IS HIGHLIGHTED.

Methods	Middlebury [46]				MPI-Sintel [47]				KITTI [49]			
	Census				Census				Census			
	w/SGM	w/SGM	w/CVF	w/CVF	w/SGM	w/SGM	w/CVF	w/CVF	w/SGM	w/SGM	w/CVF	w/CVF
	+mod.	+GCPs	+mod.	+GCPs	+mod.	+GCPs	+mod.	+GCPs	+mod.	+GCPs	+mod.	+GCPs
Initial disparity	20.43	20.43	17.64	17.64	19.54	19.54	13.82	13.82	23.50	23.50	15.21	15.21
LRC + WMF	15.61	15.61	13.80	13.80	16.81	16.81	12.37	12.37	16.77	16.77	11.74	10.74
Haeusler *et al.* [19]	12.35	14.87	12.55	11.23	12.90	13.04	10.96	10.43	10.21	9.36	10.68	9.98
Spyropoulos *et al.* [16]	13.50	15.37	12.30	10.99	12.77	11.20	11.02	10.50	9.90	8.91	10.21	9.00
Park *et al.* [7]	12.70	15.35	11.25	10.76	12.23	10.69	10.94	9.92	9.82	7.95	8.85	8.81
3DV [28]	14.20	15.78	14.32	13.23	13.48	15.15	12.48	11.86	12.49	11.56	10.89	9.10
CFA (Ours)	10.07	9.68	9.28	10.29	12.01	11.55	10.94	8.00	9.69	7.15	8.91	7.87
MCMA (Ours)	11.99	10.91	10.52	11.75	11.95	9.21	10.85	9.07	9.79	7.32	8.94	8.81
CFA+MCMA (Ours)	**9.87**	**9.19**	**9.27**	**9.32**	**11.78**	**8.61**	**10.83**	**7.49**	**9.61**	**7.12**	**8.83**	**7.80**
Optimal	7.46	4.20	6.92	3.64	7.96	6.30	8.96	5.31	7.75	2.39	7.23	2.51

best performance is achieved both components are applied, as expected. This validates the effectiveness of the proposed aggregation scheme.

D. Stereo Matching Analysis

To verify the robustness of the confidence measures, we refined the disparity map using the confidence maps estimated by several confidence measure approaches including ours. For refining the disparity maps, we used two different schemes described in Sec. 4, which are cost modulation (mod.) based optimization [7] and GCPs-based optimization (GCPs) without additional post-processing to clearly show the performance gain achieved by the proposed confidence measure. To evaluate the quantitative performance, we measured an average bad matching percentage (BMP) [46] for the MID [46], MPI [47], and KITTI [49] datasets. Table III shows the BMP for MID [46], MPI [47], and KITTI [49] dataset. For MID, since there are occluded pixels in ground truth disparity map, which we excluded, and computed the BMP only for visible pixels. The KITTI benchmark provides a sparse ground truth

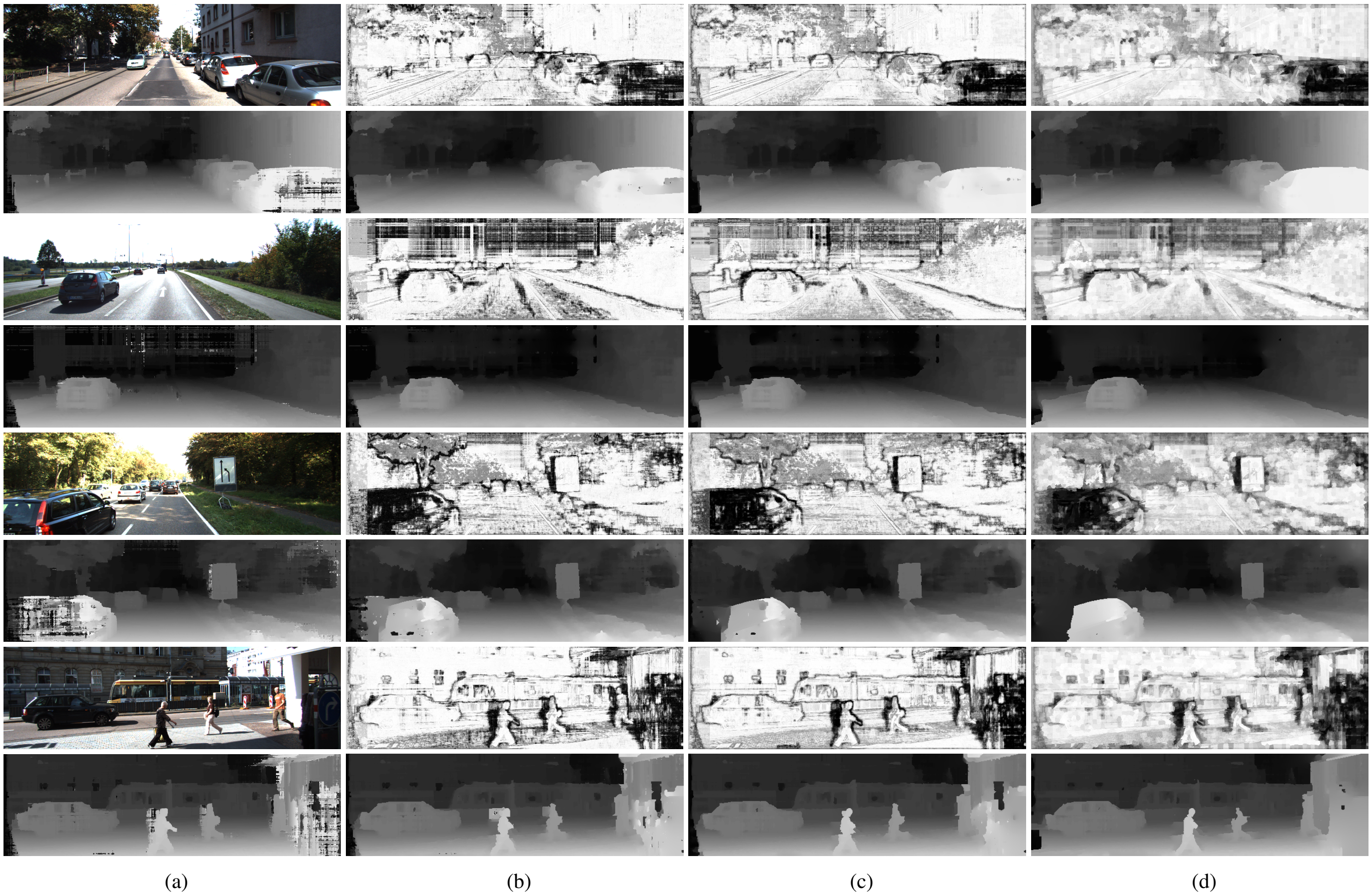


Fig. 11. The confidence maps and refined disparity maps of KITTI dataset [49] using (from top to bottom) MC-CNN [11] method. (a) color images and initial disparity maps, refined disparity maps with confidence maps estimated by (b) Haeusler *et al.* [19], (c) Park *et al.* [7], and (d) CFA+MCMA (Ours).

disparity map and we evaluated the BMP only for sparse pixels with the ground truth disparity values. The results of extensive experiments show that the proposed method achieves the lowest BMP.

Fig. 8, Fig. 9, and Fig. 10 show the disparity maps refined with the confidence maps estimated from the existing per-pixel classifiers [16], [7] and the proposed method. GCPs-based optimization was used to refine the disparity maps for the MID [46], MPI [47], and KITTI [49] datasets. It is clearly shown that the erroneous matches are reliably removed using the proposed confidence measure. For the KITTI benchmark [49], erroneous disparities usually occur in textureless regions (sky and road), and conventional approaches [7], [16] do not detect uncorrect pixels and thus they affect the matching quality of the subsequent disparity estimation pipeline. In contrast, the proposed method can detect mismatched pixels more reliably.

Furthermore, when the proposed confidence measure is combined with a cost computation method, such as MC-CNN [51], it achieves a significantly improved disparity estimation performance. Fig. 11 shows the qualitative results that refined the initial disparity maps obtained using MC-CNN cost computation for the KITTI benchmark. Table IV shows the quantitative evaluation for the KITTI benchmark using the MC-CNN based approach. The refined disparity maps with GCP-based optimization demonstrate the outstanding performance of the proposed confidence estimation method as compared to the per-pixel confidence classifier [7].

TABLE IV
THE AVERAGE BMP OF THE RESULTANT DISPARITY MAP FOR KITTI [49] DATASET. INITIAL DISPARITY IS OBTAINED WITH MC-CNN [11] BASED APPROACH. THE BMP OF THE REFINED DISPARITY MAP USING GROUND TRUTH CONFIDENCE IS MEASURED AS 'OPTIMAL'. THE RESULT WITH THE LOWEST BMP IN EACH EXPERIMENT IS HIGHLIGHTED.

Methods	KITTI [49]
	MC-CNN w/CBCA + GCPs
Initial disparity	5.6937
Spyropoulos et al. [16]	5.5361
Park et al. [7]	5.2239
CFA (Ours)	4.2621
MCMA (Ours)	4.8347
CFA+MCMA (Ours)	**4.2477**
Optimal	2.1760

For the ZED dataset, we performed a subjective evaluation only, since no ground truth disparity map exists. Fig. 12 shows the qualitative results for the ZED dataset. The refined disparity maps with cost modulation and GCPs-based optimization also support the outstanding performance of the proposed method when applied to the challenging outdoor database.

VI. Conclusion

In this study, a novel approach for learning-based confidence measure was proposed. It is assumed that the confidence features and resulting confidence maps are smoothly varying in the spatial domain. We first demonstrated that a confidence feature augmentation that imposes spatial coherency on the

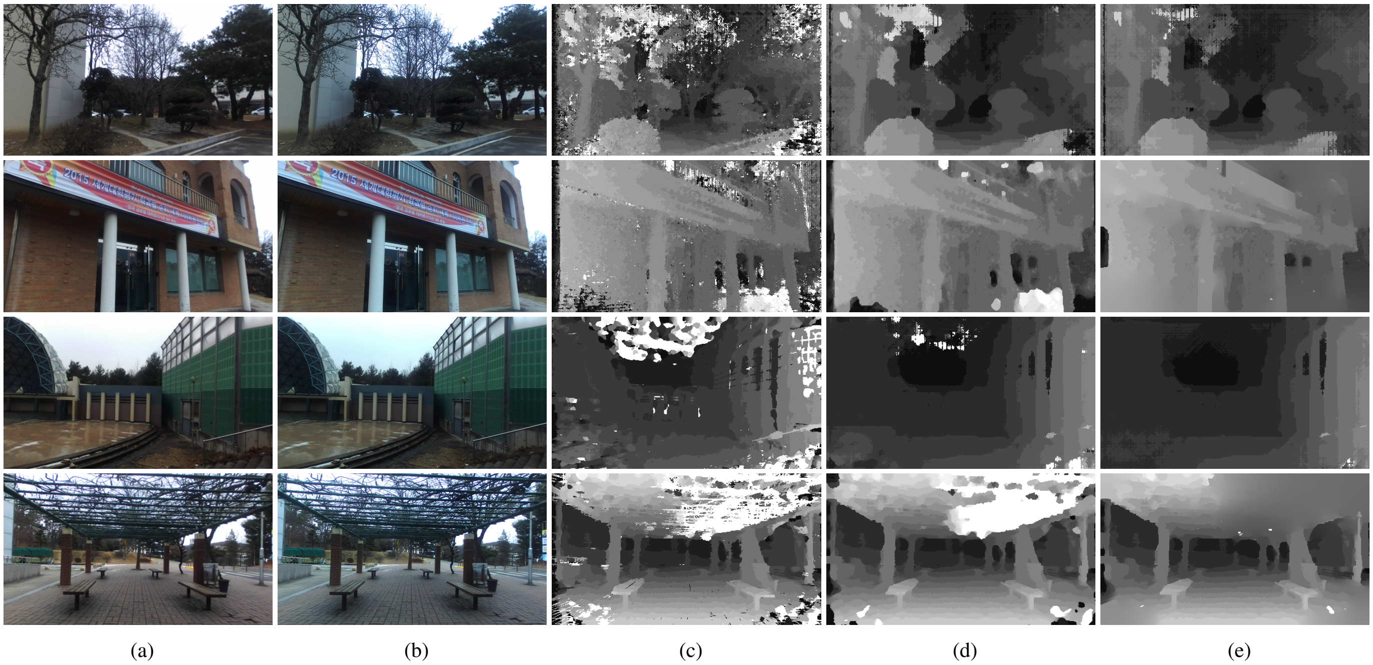


Fig. 12. The resulting disparity map of ZED dataset (from top to bottom) census-based SGM + mod., SGM + GCPs., CVF + mod., and CVF + GCPs. This dataset is captured in natural enviornments, thus frequently having illumination variations between stereo pairs. (a) left color image, (b) right color image, (c) initial disparity maps, refined disparity maps using (d) confidence measured by Park *et al.* [7], and (e) confidence measured by the proposed method.

confidence features and the resulting confidence maps can increase the performance of the confidence estimation. The confidence map was further improved through multi-scale confidence map aggregations. The stereo algorithms using the proposed confidence estimation method exhibited accurate and robust results for public datasets as well as for challenging outdoor environments. As future work, we will study the confidence feature that encodes the spatial coherency in a deep convolutional neural network framework.

References

[1] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang, "On builing an accurate stereo matching system on graphis hardware," *in Proc. IEEE Int. Conf. Comput. Vis. Work.*, pp. 467–474, Nov. 2011.

[2] M. Humenberger, C. Zinner, M. Weber, W. Kubinger, and M. Vincze, "A fast stereo matching algorithm suitable for embedded real-time systems," *Comput. Vis. Image. Understand.*, vol. 114, no. 11, pp. 1180–1202, 2010.

[3] G. Egnal and R. P. Wildes, "Detecting binocular half-occlusions: Empirical comparisons of five appraoches," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 8, pp. 1127–1133, 2002.

[4] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," *in Proc. Eur. Conf. Comput. Vis.*, pp. 151–158, May 1994.

[5] J. Kim, V. Kolmogorov, and R. Zabih, "Visual correspondence using energy minimization and mutual information," *in Proc. IEEE Int. Conf. Comput. Vis.*, pp. 1033–1040, Oct. 2003.

[6] Y. Heo, K. Lee, and S. Lee, "Robust stereo matching using adaptive normalized cross corrrelation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 4, pp. 807–822, 2011.

[7] M. Park and K. Yoon, "Leveraging stereo matching with learning-based confidence measures," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 101–109, Jun. 2015.

[8] O. Faugeras, B. Hotz, H. Mathieu, T. Vieville, Z. Zhang, P. Fua, E. Theron, L. Moll, G. Berry, J. Vuillemin, P. Bertin, , and C. Proy, "Real time correlation-based stereo: Algorithm, implementations and applications," Inria, Tech. Rep., 1993.

[9] D. Min and K. Sohn, "Cost aggregation and occlusion handling with wls in stereo matching," *IEEE Trans. Image Process.*, vol. 17, no. 8, pp. 1431–1442, 2008.

[10] S. Kim, B. Ham, B. Kim, and K. Sohn, "Mahalanobis distance cross-correlation for illumination invariant stereo matching," *IEEE Trans. Circ. Syst. Vid. Techn.*, vol. 24, no. 11, pp. 1844–1859, 2014.

[11] J. Zbontar and Y. Lecun, "Computing the stereo matching cost with a convolutional neural network," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 1592–1599, Jun. 2015.

[12] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 328–341, 2008.

[13] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 3017–3024, Jun. 2011.

[14] G. Egnal, M. Mintz, and R. Wildes, "A stereo confidence metric using single view imagery with comparison to five alternative approaches," *Image. Vis. Comput.*, vol. 22, no. 12, pp. 943–957, 2004.

[15] L. Wang and R. Yang, "Global stereo matching leveraged by sparse ground control points," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 3033–3040, Jun. 2011.

[16] A. Spyropoulos, N. Komodakis, and P. Mordohai, "Learning to detect ground control points for improving the accuracy of stereo matching," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 1621–1628, Jun. 2014.

[17] P. Mordohai, "The self-aware matching measure for stereo," *in Proc. IEEE Int. Conf. Comput. Vis.*, pp. 1841–1848, Sep. 2009.

[18] D. Kong and H. Tao, "A method for learning matching errors in stereo computation," *in Proc. Brit. Mach. Vis. Conf.*, p. 2, Sep. 2004.

[19] R. Haeusler, R. Nair, and D. Kondermann, "Ensemble learning for confidence measrues in stereo vision," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 305–312, Jun. 2013.

[20] A. Fusiello, V. Roberto, and E. Trucco, "Efficient stereo with multiple windowing," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, p. 858, Jun. 1997.

[21] H. Hirschmuller, P. Innocent, and J. Garibaldi, "Real-time correlation-based stereo vision with reduced border errors," *Int. J. Comput. Vis.*, vol. 47, no. 1–3, pp. 229–246, 2002.

[22] G. Ulrike, "Variable importance assessment in regression: Linear regression versus random forest," *The Americ. Statistic.*, 2012.

[23] L. B. Statistics and L. Breiman, "Random forests," *Mach. Learn.*, vol. 63, no. 4, pp. 5–32, 2001.

[24] A. Liaw and M. Wiener, "Classification and regression by random forest," *R news*, vol. 2, no. 3, pp. 18–22, 2002.

[25] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, no. 1–3, pp. 7–42, 2002.

[26] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient belief propagation for early vision," *IJCV: Int. J. Comput. Vis.*, vol. 70, no. 1, pp. 41–54, 2006.

[27] P. F. Felzenszwalb and R. Zabih, "Dynamic programming and graph algorithms in computer vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 4, pp. 721–740, 2011.

[28] R. Gouveia, A. Spyropoulos, and P. Mordohai, "Confidence estimation for superpixel-based stereo matching," *in Proc. IEEE Int. Conf. 3D Vis.*, pp. 180–188, Oct. 2015.

[29] K. Yoon and I. Kweon, "Distinctive similarity measure for stereo matching under point ambiguity," *Comput. Vis. Image. Understand.*, vol. 112, no. 2, pp. 173–183, 2008.

[30] X. Hu and P. Mordohai, "A quantitative evaluation of confdience measures for stereo vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2121–2133, 2012.

[31] D. Pfeiffer, S. Gehrig, and N. Schneider, "Exploiting the power of stereo confidences," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 297–304, Jun. 2013.

[32] A. Seki and M. Pollefeys, "Patch based confidence prediction for dense disparity map," *in Proc. Brit. Mach. Vis. Conf.*, vol. 10, Sep. 2016.

[33] K. He, J. Sun, and X. Tang, "Guided image filtering," *in Proc. Eur. Conf. Comput. Vis.*, pp. 1–14, Sep. 2010.

[34] P. Dollar and C. L. Zitnick, "Structured forests for fast edge detection," *in Proc. IEEE Int. Conf. Comput. Vis.*, pp. 1841–1848, Jun. 2013.

[35] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, "Support vector machine learning for interdependent and structured output spaces," *in Proc. ACM Int. Conf. Mach. Learn.*, p. 104, Jul. 2004.

[36] B. Taskar, V. Chatalbashev, D. Koller, and C. Guestrin, "Learning structured prediction models: a large margin approach," *in Proc. ACM Int. Conf. Mach. Learn.*, pp. 896–903, Aug. 2005.

[37] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 839–846, Jan. 1998.

[38] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, 2012.

[39] C. Biernacki, G. Celeux, and G. Govaert, "Assessing a mixture model for clustering with the integrated completed likelihood," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 7, pp. 719–725, 2000.

[40] T. K. Moon, "The expectation-maximization algorithm," *IEEE Sign. process. magaz.*, vol. 13, no. 6, pp. 47–60, 1996.

[41] Q. Yang, L. Wang, R. Yang, H. Stewenius, and D. Nister, "Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 3, pp. 492–504, 2009.

[42] E. S. Gastal and M. M. Oliveira, "Adaptive manifolds for real-time high-dimensional filtering," *ACM Trans. Graph.*, vol. 31, no. 4, p. 33, 2012.

[43] I. Jolliffe, *Principle component analysis*. Wiley Online Library, 2002.

[44] I. Wald and V. Havaran, "On building fast kd-trees for ray tracing, and on doint that in o(n log n)," *in IEEE Symp. Int. Ray Trac.*, pp. 61–69, Sep. 2006.

[45] J. Sun, N. N. Zheng, and H. Y. Shum, "Stereo matching using belief propagation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 7, pp. 787–800, 2003.

[46] [Online] http://vision.middlebury.edu/stereo/.

[47] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," *in Proc. Eur. Conf. Comput. Vis.*, pp. 611–625, Oct. 2012.

[48] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," [Online] http://www.cvlibs.net/datasets/kitti/.

[49] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," *in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 3061–3070, Jun. 2015.

[50] L. Xu, Q. Yan, and J. Jia, "A sparse control model for image and video editing," *ACM Trans. Graph.*, vol. 32, no. 6, p. 197, 2013.

[51] [Online] http://github.com/jzbontar/mc-cnn.

[52] K. Zhang, J. Lu, and G. Lafruit, "Cross-based local stereo matching using orthogonal integral images," *IEEE Trans. Circ. Syst. Vid. Techn.*, vol. 19, no. 7, pp. 1073–1079, 2009.

[53] [Online] http://kr.mathworks.com/help/stats/treebagger-class.html.

[54] [Online] https://www.stereolabs.com/.