Semantic Segmentation-based Stereo Reconstruction with Statistically Improved Long Range Accuracy

Vlad-Cristian Miclea and Sergiu Nedevschi

***Abstract*— Lately stereo matching has become a key aspect in autonomous driving, providing highly accurate solutions at relatively low cost. Top approaches on state of the art benchmarks rely on learning mechanisms such as convolutional neural networks (ConvNets) to boost matching accuracy.**

We propose a new real-time stereo reconstruction method that uses a ConvNet for semantically segmenting the driving scene. In a ”divide and conquer” approach this segmentation enables us to split the large heterogeneous traffic scene into smaller regions with similar features. We use the segmentation results to enhance Census Transform with an optimal census mask and the SGM energy optimization step with an optimal P_1 penalty for each predicted class. Additionally, we improve the sub-pixel accuracy of the stereo matching by finding optimal interpolation functions for each particular segment class. In both cases we propose new stochastic optimization methods based on genetic algorithms that can incrementally adjust the parameters for better solutions. Tests performed on Kitti and real traffic scenarios show that our method outperforms the accuracy of previous solutions.

I. INTRODUCTION

Over the last two decades stereo vision has proven to be a viable, low-cost method for obtaining depth information in various environments and for diverse applications. Even tough lately its ubiquitous usage has been surpassed by the apparition of Velodyne Lidar its capabilities still make the stereo vision sensor an important tool for computer vision. Although extremely accurate for medium-range depth distances, Velodyne is expensive and suffers from sparse results which makes them unreliable to some extent in the far distance setting (more than 50m).

The classic taxonomy of stereo reconstruction algorithms separates them into two categories: global or local methods. Local approaches evaluate the disparity relying on a similarity criterion applied over small (generally maximum 5x5) support windows. Global methods evaluate the disparity of all pixels in an image as a whole by optimizing a global energy function. A boost in performance of local algorithms has appeared lately in conjunction with convolutional neural networks (ConvNets) [1], [2] which are best suited for optimizing the similarity criterion.

From a different perspective, stereo reconstruction algorithms can be seen as either discrete or continuous. According to this criteria, algorithms in continuous space have proven more accurate with increased density, at larger computational cost. On the other hand, the discrete algorithms offer real-time performance, with pixel-wise comparable results. However when sub-pixel accurate results are required (for long range accuracy), they suffer from ”pixel-locking effect”: an over-crowding of disparities towards integer values.

The most significant algorithm in the discrete category is the Semi-Global Matching (SGM) [3] [4]. This method stays at the edge between local and global solutions since it performs multiple 1D energy optimizations on the image, in order to approximate a 2D optimization. The energy optimization is based on a data correlation term (usually Census Transform – CT [5]) and a smoothness constraint enforced by the use of two penalities. Results with major dense stereo correspondence datasets (Middlebury [6], Kitti [7]) reveal this method as good compromise between speed and accuracy.

This paper presents an improved Census-based SGM solution that exploits ConvNets for increasing the algorithm’s density and accuracy. The network is not used directly for finding a similarity measure but rather as a preprocessing technique for performing an initial semantic segmentation of the scene. We also propose three new genetic algorithms, two for pixel-level enhancement of SGM by optimizing the census mask and the P_1 penalty, and one for generating the optimal sub-pixel interpolation function. The paper starts with presenting the state of the art in segmentation census-based and sub-pixel accurate stereo solutions. The next Section describes the workflow of the proposed method, the ConvNet-based image segmentation and the newly introduced pixel-level genetic algorithms. In Section 4 we discuss in detail the enhancements we propose for the sub-pixel part of the algorithm. Section 5 presents the accuracy improvements produced by gradually introducing each new method. Moreover, it depicts results obtained in other driving scenarios. Finally, we conclude the paper in Section 6.

II. RELATED WORK

A. Segmentation-based Stereo

Several methods that combine image segmentation with stereo have been developed during the last years. Such algorithms try to increase disparity accuracy by using segmentation as a post processing step. The authors of [8] extract confident disparity pixels and propose a plane fitting-based segmentation to fill the disparity holes. The same goal is followed by the method of [9], but is using super-pixels as a method to group similar pixels. The problem with all these super-pixel based implementations is the increased computational effort making them not viable for real-time usage.

The authors are with the Department of Computer Science, Technical University of Cluj-Napoca, Cluj-Napoca, Romania, E-mails: Vlad.Miclea@cs.utcluj.ro, Sergiu.Nedevschi@cs.utcluj.ro

More recently semantic segmentation has become more and more reliable for confidently highlight scene objects. The authors in [10] rely on the similarity given by specific object structures to fill sparse disparity estimates. In the recent article [11] the authors proposed to apply semantic segmentation over their stixel-based stereo method to enrich the scene information and obtain more reliable results.

B. *Census-based methods*

A good data correlation term in stereo matching is the key in obtaining accurate results. Most approaches nowadays prefer to focus on non-parametric transforms (Matrix rank transform, Birchfeld Tomasi [12]) instead of using intensity-based metrics (SAD, SSD, NCC). Census Transform [5] for instance stands out among these methods not only due to its non-parametric properties, but also because it can be easily implemented on any type of hardware.

New types of Census Transforms have been developed [13], [14], producing better results at smaller costs. These methods are improved by either using a Center-Symmetric Census and/or by choosing larger, sparse census windows. A mask is applied over these windows in order to select the pixels containing the most significant information. Notable census masks are the Star [15], the Center-Symmetric [14] and the Chessboard [14]. In our previous work [16] we show that better census masks can be created through stochastic optimization and we propose a new method that can obtain improved results.

C. *Long-range stereo for discrete case*

Various approaches for sub-pixel improvement of stereo algorithms have been proposed during the last decade. A relevant study is shown in [17], which accounts for the best methods in both discrete and continuous domain. Sub-pixel improvement techniques based on a mathematical model [18] are easily being correlated with simple local algorithms such as SAD or SSD. For more complex algorithms (such as SGM) good sub-pixel compensation techniques have been developed experimentally. [19], [20], or [21] use function fitting as a statistically-based process for finding a function that can alleviate the pixel-locking effect.

The problem with all these methods is that a function is generally fitted to optimally operate on fronto-parallel surfaces. This results in a performance degradation for slanted or irregular surfaces. Although giving globally better results, LUT-based methods [22] do not solve entirely the situation of non-homogeneity in the scene.

III. Methodology and pixel-level improvements

A. *Overall Architecture*

Since most of the aforementioned methods suffer from a lack of parameter globalization, we propose to divide the scene into homogeneous regions. In the initial step of the algorithm we train a convolutional neural network for accurate pixel-wise scene segmentation. Then, for each class we determine an optimal census mask, an optimal SGM penalty P_1 and an optimal sub-pixel interpolation function. An overview of the proposed stereo method can be seen in Fig 1.

B. *Semantic segmentation of the image*

As previously mentioned, combining image segmentation with stereo is not a new trick. Latest approaches for semantic segmentation [23] nowadays use learning mechanisms such as boosting [24] or convolutional neural networks [25], [26] and they can obtain more than 95% in pixel accuracy classification. Among top methods we can find FCN [27], that uses the a modified VGG ConvNet with 16 layers, producing fairly good results at good speed (6-7 fps). Although it can classify each driving scene pixel into 35 classes, we slightly modify the architecture to account for a smaller number of classes. The proposed segment classes are:

1) The road surface area
2) The sidewalk and the side terrain
3) Large road obstacles like cars, trucks etc. which are generally fronto-parallel to the cameras
4) Small road obstacles like pedestrians or traffic signs
5) Large side objects like buildings, walls or fences
6) Irregular shapes such as vegetation and trees
7) The sky region above features from other classes. This class will not be used for stereo for obvious reasons (disparity is 0).

Although a larger number of classes might provide more a-priori information, observations revealed by exhaustive testing show that an increased granularity for segmentation can introduce processing difficulties (especially for segments composed by fewer pixels) and may lead to increased computational costs.

C. *Pixel level genetic algorithms*

As each segment class provides enough information about the surface structure, we further apply optimizations in ”divide and conquer” manner. Therefore, according to each segment class we can determine:

- the optimal census mask. As presented in [16], a viable census masks usually covers a surface of maximum 15x15 pixels, giving enough information and allowing for maximum 32 pixels to be selected. We present the methodology for finding a segment-dependent optimal census mask in Algorithm 1. If we account for more than 32 bits, the processing time increases beyond the real-time requirement.
- the optimal P_1 penalty for SGM energy minimization step. As P_1 acknowledges for small disparity changes, we can assume that these changes can also be dependent on the surface. Therefore, we propose a new strategy (Algorithm 2) that accounts for these cases and establishes an optimal P_1 for each semantic segment. On the other hand, P_2 penalty is generally used for large disparity changes. Since the semantic segmentation algorithm accurately separates the traffic image into rather homogeneous classes, we can assume that large disparity disruptions entail segment changes. Moreover, it has been previously shown [28] that SGM

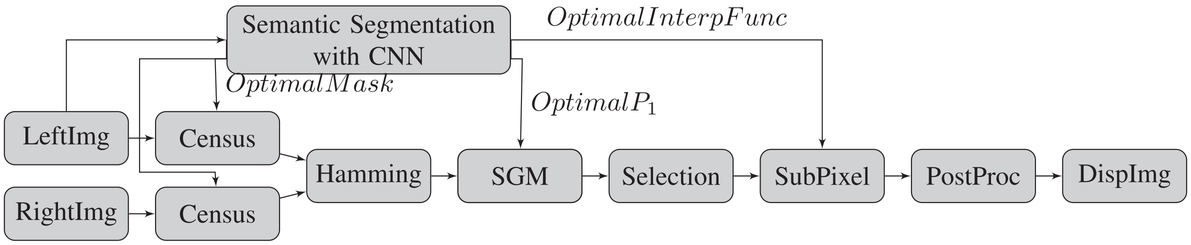


Fig. 1: Workflow of the Proposed Algorithm

produces best results when P_2 is adapted to the intensity surface. Therefore, P_2 is not accounted for stochastic optimization.

Algorithm 1 Algorithm for Optimal Census Mask

```
procedure GENETIC ALG. FOR CENSUS
    for all segments do
        initialize population(0)
        $d_{CT} \leftarrow$ apply CT(population(0))
        population.fitness(0) $\leftarrow err(d_{CT}, d_{GT})$
        repeat
            perfrom selection, crossover and mutation on population(i)
            partially initialize population(i+1)
            population(i+1) $\leftarrow$ population_mut(i) + population(i+1)
            $d_{CT} \leftarrow$ apply CT(population(i+1))
            population.fitness(i+1) $\leftarrow err(d_{CT}, d_{GT})$
        until i=finalGeneration
    end for
end procedure
```

Algorithm 2 Algorithm for Optimal SGM Penalty P_1

```
procedure GENETIC ALG. FOR $P_1$
    for seg = 1 to all segments do
        initialize population(0) with $P_1(seg)$
    end for
    $d_{SGM} \leftarrow$ apply SGM(population(0))
    population.fitness(0) $\leftarrow err(d_{SGM}, d_{GT})$
    repeat
        perfrom selection, crossover and mutation on population(i)
        for seg = 1 to all segments do
            partially initialize population(i+1) with $P_1(seg)$
        end for
        population(i+1) $\leftarrow$ population_mut(i) + population(i+1)
        $d_{SGM} \leftarrow$ apply SGM(population(i+1))
        population.fitness(i+1) $\leftarrow err(d_{SGM}, d_{GT})$
    until i=finalGeneration
end procedure
```

In both these algorithms the operator (+) defines a concatenation between the newly generated and the preserved populations. Details about the selection, crossover and mutation steps of the algorithm are extensively highlighted in [16].

IV. Sub-pixel enhancements

A. Sub-pixel Interpolation Theory

To further improve the accuracy of the stereo matching we additionally focus on the fractional part of the disparity. Standard methods estimate the sub-pixel disparity by adapting the integer value with a sub-unitary amount, such as:

$$d_{SubPx} = d_{Int} + f(c_{d-1}, c_d, c_{d+1}) \tag{1}$$

where d_{Int} is the integer disparity value, c_d is the cost at the chosen disparity, while c_{d-1} and c_{d+1} represent the neighboring costs (taken from the cost volume generated using SGM).

Previous approaches have shown that f can be modeled as depending on only one parameter (instead of 3) by:

- A translation of costs to the origin of c_d on the real axis
$$\begin{aligned} l_d &= c_{d-1} - c_d \\ r_d &= c_{d+1} - c_d \end{aligned} \tag{2}$$
- A correlation of l_d and r_d based on the observations presented in [20]:
$$x = l_d / r_d \tag{3}$$
with sub-pixel distribution being symmetric with respect to integer values.

Therefore, reformulating the sup-pixel disparity (1) according to (2) and (3), we get:

$$d_{SubPix} = \begin{cases} d_{Int} - 0.5 + f(x) & \text{if } l_d \leq r_d \\ d_{Int} + 0.5 - f(1/x) & \text{otherwise} \end{cases} \tag{4}$$

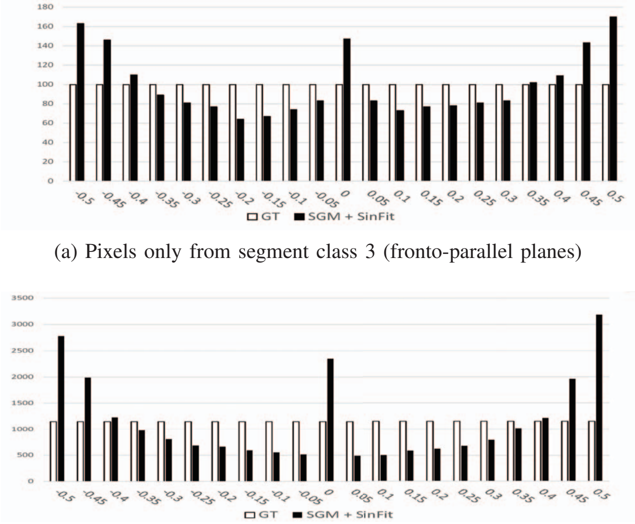
B. Function Fitting

To model a function such that it gives you good distributions is a laborious process, known as *function fitting*. Previous approaches have shown that resulting functions must meet the following properties:

- it is defined on the interval $[0, 1]$, with values in $[0, 0.5]$;
- it is monotonically increasing.

Literature highlights lots of good candidates, starting from simple functions such as parabola (5):

$$f(x) = \frac{x}{x+1} \tag{5}$$



(a) Pixels only from segment class 3 (fronto-parallel planes)

(b) Pixels from all segments

Fig. 2: GT is the expected disparity distribution, SGM + SinFit are the histograms obtained with the SinFit correction proposed in [20]

or equiangular (symmetric V)(6):

$$f(x) = 0.5 \times x \tag{6}$$

and continuing to more complex ones, such as SinFit (7) or MaxFit(8):

$$f(x) = 1/2 \times (sin(x \times \frac{\pi}{2} - \frac{\pi}{2}) + 1) \tag{7}$$

$$f(x) = max((1 - cos(x \times \frac{\pi}{3})), (x^4 + x)/4) \tag{8}$$

Although these functions work fine on fronto-parallel planes, their behavior becomes unsteady in case of more complex surfaces.

A clear understanding of the sub-pixel behavior for a particular interpolation function can be seen by shifting the view from 3D space to a histogram representation. The histogram bins are considered classes of points corresponding to equal depth intervals in the range [-0.5, 0.5]. Each image pixel from the scene is classified according to its fractional part of the disparity in one of the histogram bins. In Fig 2a we show a histogram representation of points that resulted after applying SGM + SinFit (one of the best sub-pixel methods) with respect to the Ground Truth. The GT points are randomly chosen, but for representation purposes we force equal number of points for all GT classes. The histogram shows that if we account only for pixels in segment class 3 (fronto-parallel planes), the point re-distribution is not that far from GT. On the other hand, if we account for pixels in all classes (Fig 2b), besides the fact that pixel-locking effect is still present, the overall re-distribution is distorted.

For both these figures it can easily be seen that more complex functions have to be developed and each function must be selected according to the properties of each particular segment.

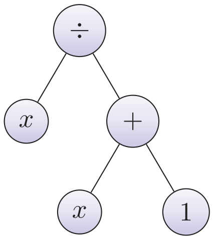


Fig. 3: Parabola as expression tree

C. Sub-Pixel level Genetic Algorithm

As previously shown we are looking for functions that have an increased complexity, so they produce convincing results for each particular segment class. Since the ”by hand” function fitting methodology presented so far is laborious and time-consuming we approach this step using genetic algorithms which search for the best set of functions (individuals) that correctly redistributes image points at sub-pixel level.

In this context each function is described as a binary expression tree, having as operators the following set of nodes which allow for increased flexibility:

- Unary operators: $sin, cos, tan, arcsin, arccos, arctan, log, sqrt, min, max$
- Binary operators: $+, -, *, /$
- $EMPTY$ - for uncompleted branches

The operands of each generated function are situated on the leaves of each binary tree and represent a sufficient subset of possible:

- Constants: $1, 2, 5$, π, $\pi/2$, e (Euler constant)
- Variables: $x = l_d/r_d$ or $x_1 = l_d, x_2 = r_d$

For instance, the expression in equation 5 (parabolic function) can be seen as the following expression tree (Fig 3).

Algorithm 3 describes the way in which an optimal interpolation function is found. Although sub-pixel interpolation theory related to SGM [20], [19] shows that $x = l_d/r_d$ can be used as a single variable, the algorithm permits for both l_d and r_d to be used free variables. In terms of function complexity, this is given by the tree depth, established as either 5 or 6. Thus, the most complex functions will be of maximum 32 or 64 operands (complete binary expression tree).

Remodeling an interpolation function can be done by:

- **Crossover**: Considering 2 functions $f_1(x)$ and $f_2(x)$ presented as expression trees, randomly selected branches (together with their descendants) are interchanged between f_1 and f_2. It results in parts of expressions being changed;
- **Mutation**: In order to ensure evolution, randomly selected nodes are periodically changed inside the expression tree.

The fitness for each population member is computed in two steps (mathematically described in procedure *Fitness Computation* in Algorithm 3):

Algorithm 3 Algorithm for Optimal Sub-pixel Interpolation Function

```
procedure POINT EXTRACTION FOR SUB-PIXEL
    for $sp\_cls$ = 1 to $class\_no$ do
        select randomly k GT points in $sp\_cls$
        extract $l_d$ and $r_d$ costs at selected points
    end for
end procedure
procedure FITNESS COMPUTATION
    for $sp\_cls$ = 1 to $sp\_class\_no$ do
        evaluate f(x) for all k values
        $med\_val(sp\_cls) \leftarrow median(f(x))$
    end for
    population(0).genome(i).fitness $\leftarrow$ $max\_err(med\_val(sp\_cls), GT(sp\_cls))$
end procedure
procedure GENERIC ALGORITHM FOR SUB-PIXEL
    apply SGM and save $l_d$ and $r_d$ for all points
    for all segments do
        perform Point Extraction for Sub-Pixel
        perform Fitness Computation for all genimes in popuation(0)
        repeat
            remodel f(X) by crossover and mutation
            on randomly selected genomes from population(i)
            select population_best(i)
            partial initialize population(i+1)
            population(i+1) $\leftarrow$ population_best(i) + population(i+1)
            perform Fitness Computation for all genomes in popuation(i+1)
        until i=finalGeneration
    end for
end procedure
```

1) Firstly we compute the representative value per sub-pixel class. This is done by applying the interpolation function for all points in each sub-pixel class and then finding the median value per class;
2) Secondly, we compute the difference between the sub-pixel representative of each class (the median) and its corresponding ground truth value. Finally, the maximum absolute difference among all sub-pixel classes will be computed as fitness.

V. EXPERIMENTAL DATA

A. Data Analysis

1) Pixel level evaluation methodology: We use the classical evaluation methodology for stereo vision algorithms by testing various census masks and various SGM penalties on real traffic images. [7] is the main dense stereo correspondence benchmark for driving scenarios, so we use its 2015 dataset [29] for training and then testing our method. The classification criteria is the percent of misclassified pixels with respect to the nonzero pixels in the ground truth. We use 80 images for training – census and P_1 optimization. We use the entire test set of 200 images for evaluation. We perform two types of tests: Initially we show the results obtained by only using the census transform (without any aggregation or optimization). We then present the results obtained by integrating the optimization mechanism into SGM. Since the results in [14] clearly show the superiority of center-symmetric censuses over the regular ones, all evaluated methods employ just the center-symmetric case.

2) Sub-pixel level evaluation methodology: In our previous works [21], [22] we centered the evaluation process on the histogram domain, comparing distributions generated by new sub-pixel methods with ground truth (GT) distributions. Consequently, we introduced two new metrics, accuracy percent (global accuracy percentage) and peak mismatching (locally, the worst-case histogram mismatch). However, in this work we leave out this intermediate representation and we evaluate the maximum error produced by each function directly by applying the methodology presented for fitness function computation. Solutions with more than 90% accuracy were developed for each particular segment class.

We train on a randomly chosen subset of points obtained from the entire dataset so we are not restricted to fronto-parallel/tilted planes([20], [19], [30], [21]). As the proposed method intends to alleviate errors at sub-pixel level, the training phase will be performed only on points correctly matched at pixel value.

To determine the percentage of erroneous matches, the Kitti benchmark employs a minimum threshold of two pixels. Despite this, ground truth images (16 b/px, Velodyne-based) are good enough for our sub-pixel evaluation methodology so we were able to perform tests on this set.

B. Results

1) Results at pixel level on Kitti dataset: We test the following masks:

- Star – the pattern proposed by [15]. This is a symmetric pattern containing 32 pixels inside a 9×9 window;
- Center-avoiding – the pattern introduced in [14]. This pattern selects pixels that are situated at a 2-3 pixel distance from the center, neither too far, nor too close;
- Dense – This pattern is the most simple one. It accounts for all the pixels in the image. Because of its proportional spreading to size feature, it gives larger processing times for larger windows. We considered here a 7×7 window;
- GA – This is the method based on the genetic algorithm proposed in [16], selecting an optimal census mask for the entire set of images. It selects optimally 32 pixels inside a 11×11 window so that resulting cost is represented on 32 bits.
- GA + Seg – The optimal masks and optimal P_1 are given by the proposed genetic algorithms 1 and 2. This contains a set of 6 census masks, each of them containing specific (max 32) pixels inside a 11×11 window and 6 penalties, which are constant for the

TABLE I: Average Error percent for Census-only and SGM obtained with various Census Masks (error thresh $T = 3$)

	Seg1		**Seg2**		**Seg3**		**Seg4**		**Seg5**		**Seg6**		**Total**	
Method	Census	SGM	Census	SGM	Census	SGM	Census	SGM	Census	SGM	Census	SGM	Census	SGM
Star	89.09%	6.84%	65.51%	15.75%	74.59%	12.01%	65.15%	25.51%	71.14%	25.44%	84.25%	15.46%	82.46%	11.26%
CenterAv	92.87%	11.61%	70.23%	15.48%	82.62%	16.39%	67.59%	25.11%	74.80%	25.45%	89.67%	14.93%	87.79%	11.98%
Dense	83.45%	5.86%	72.16%	16.15%	71.63%	11.58%	62.07%	23.14%	69.10%	25.00%	89.43%	17.67%	79.64%	10.79%
GA	76.49%	4.29%	53.02%	14.17%	60.80%	13.06%	64.32%	21.41%	**63.61%**	23.78%	72.24%	12.88%	70.40%	9.82%
GA+Seg	**63.02%**	**3.92%**	**51.32%**	**14.17%**	**60.32%**	**12.74%**	**57.19%**	**20.56%**	63.62%	**23.15%**	**71.60%**	**12.34%**	**62.22%**	**8.69%**

specific class, where the lowest is 7 and the highest is 35.

The results obtained at pixel-level are presented in Table I. A first observation is that since more than 50% of the pixels belong to the road surface, algorithms that lead on that specific surface (Dense, and GA-based) top the overall ranking. Another remark is that although regular (dense) CT thrives on regular surfaces – fronto parallel and road, it behaves poorly on irregular objects such as vegetation and terrain.

All in all, we can notice that our newly introduced approach ranks first in this classification, outperforming the non-segmented GA approach with almost 10% for the Census-only case, and the other methods by more than 17%. However, this margin strongly decreases when we introduce the energy minimization term. This happens because the SGM energy minimization compensates for the lack of correlation accuracy. An additional uncertainty is added by the inherent segmentation errors at object interactions so we can say that our method would work even better with improved semantic segmentation techniques [24].

2) Results at sub-pixel level on Kitti dataset: The subset of points is generated using the same randomized selection as described in the training phase, therefore the points are different from one subset to another even when selected from identical images. The tested sub-pixel enhancement methods are:

- SymmetricV interpolation function
- Parabola interpolation function
- SinFit – sinusoidal function proposed in [19]
- MaxFit – proposed in [21]
- LUT correction over results with SymmetricV proposed in [22]
- GA_{SP} – optimal interpolation function for all images
- $GA_{SP} + Seg$ – optimal interpolation functions for each segment. The following functions have been used:

$$f_{S1}(x) = sin(x^4 \times \pi/2) \tag{9}$$

$$f_{S2}(x) = 1/2 \times ((x + \sqrt{x}) - (2 - x)^2) \times \sqrt{x} \times (3x - 2)^3 \tag{10}$$

$$f_{S3}(x) = \frac{sin(x * \pi/2)}{\frac{2+x}{3}} \times \frac{1}{x + 1} \tag{11}$$

$$f_{S4}(x) = 1/2 \times (sin(x \times \frac{\pi}{2} - \frac{\pi}{2}) + 1) \tag{12}$$

$$f_{S5}(x) = (\frac{2\sqrt{x}}{x^2 + 1} + sin(x^2\pi)) \times \frac{1}{x + 1)} \tag{13}$$

$$f_{S6}(x) = (\sqrt{x} - \frac{x^4}{2})((2 - x)2x - 2 + \sqrt{x})(4 - 2x - x^4) \tag{14}$$

A first observation is that some of these functions do not entirely respect thes aforementioned properties. For example the function in equation (14) defined on the interval [0,1] with values in [0, 0.5] is still continuous, but not monotonic. This is a clear consequence of sub-pixel distribution difference between simple (frontal) and complex surfaces.

Tests performed over Kitti images (Table II) reveal higher sub-pixel errors for parabola function. This indicates the presence of a strong pixel-locking effect in this case. The error is reduced for the SymmetricV, SinFit and MaxFit, especially for Segments 3 and 4, that correspond to simple fronto-parallel surfaces (cars, pedestrians etc). Both LUT-based and GA_{SP} approaches show accurate global behavior, but still give poor results for Segments 2, 5 and 6 – complex surfaces.

The results obtained in Segment 4 (fronto-parallel, slim objects) stand out. In this case, previously presented solutions (SinFit) outperformed the GA-generated solution with a small margin (because the function found during the training phase obtained only 90% in accuracy). Therefore we selected SinFit as best candidate for this particular segment. Consequently, $GA_{SP} + Seg$ approach will always outperform (or at least produce similar results to) methods without surface knowledge.

3) Execution time and possible enhancements: In terms of execution performance, the additional segmentation takes an additional time of 180 ms. If we use the approach of [24] the semantic segmentation time can be reduced to only 40 ms, with similar accuracy and preserving the real-time constraint. Table III presents details about execution time required by each step on a Intel i5 CPU with 4 cores @ 3.30 GHz. RGB-to-grayscale, WTA and left-right checking execution times are included in the presented steps.

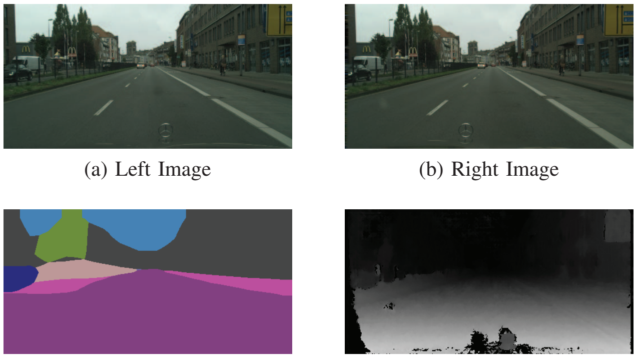
Scene segmentation comes with other benefits than just delivering relevant information about surface structure. For instance, a cross-based aggregation that accounts for scene surfaces can easily be implemented in conjunction with our optimal census generation algorithm. Besides this, there are other stereo matching tricks that can be designed with respect to segmentation (dealing with slanted-planes, post-processing for hole filling etc). Therefore, we consider that a fair comparison with state of the art algorithms on Kitti testing dataset would be fair only if we plug-in some (or

TABLE II: Average Error of sub-pixel techniques on Kitti Images (error thresh $T = 1$)

Method	**S1**	**S2**	**S3**	**S4**	**S5**	**S6**	Total
SymmetricV	19.10%	44.71%	40.61%	44.37%	49.88%	34.09%	30.83%
Parabola	20.65%	46.72%	42.22%	46.2%	51.92%	34.88%	33.29%
SinFit	20.09%	44.97%	40.02%	**41.62%**	50.28%	34.59%	31.20%
MaxFit	19.47%	45.20%	40.07%	42.02%	50.28%	34.50%	31.08%
LUT SymmV	19.10%	44.57%	40.22%	41.88%	50.22%	34.07%	30.77%
GA	19.02%	43.76%	40.27%	41.73%	49.77%	34.59%	30.52%
GA + seg	**18.20%**	**41.29%**	**37.93%**	**41.62%**	**45.27%**	**31.25%**	**27.67%**

TABLE III: Execution time per each stage (ms) for Kitti images (375x1248)

Method	**Seg**	**Census**	**EnergyMin**	**Sub-Pixel**	**PostProc**	**Total**
Proposed	180	40	180	15	4	419
Regular SGM	-	38	170	10	4	222


(a) Left Image

(b) Right Image

(c) Segmentation Image

(d) Disparity Image

Fig. 4: Results obtained on images from CityScapes dataset

all) of these mechanisms for increased disparity reliability. However, introducing these additional refinements would be outside the scope of this paper so we leave this for future work.

4) Evaluation on Traffic images: Our method requires color traffic images (for semantic segmentation) with sub-pixel accurate disparity ground truths. To the best of our knowledge Kitti2015 is the only benchmark that fulfills these requirements. While stereo datasets such as Middlebury [31] contain only indoor scenarios, segmentation datasets such as CityScapes [23] or Pascal [32] lack in sub-pixel accurate depth ground truths. We choose to show the results of our method on images from CityScapes dataset due to its traffic scenarios. Figure 4 depicts a)-b) the left and right images c) the segmented image and d) disparity map obtained with our method. Although we can not present numerical results for this set, the resulted disparity is dense and accurate, while the road smoothness indicates the absence of pixel locking.

VI. CONCLUSIONS

Learning methods such as ConvNets are becoming more and more popular in the stereo reconstruction domain. Instead of using such methods for computing a similarity measure, we use a ConvNet for semantically segmenting the driving scene, which is then used as pre-processing for stereo. For each particular segment we compute an optimal census mask and an optimal SGM P_1 penalty in order to enhance the reliability of the pixel-level disparity map. We also propose new sub-pixel interpolation functions for each segment class. Three new genetic algorithms are proposed for these optimizations. We performed multiple tests on different types of data with best positive results for the proposed approach.

We intend to continue our work by applying this methodology to other discrete stereo reconstruction solutions. Moreover, we plan to extend this method by combining ConvNet-based similarity measures with our optimal Census, by improving the reconstruction on slanted planes and by using semantic segmentation for additional post-processing improvements.

ACKNOWLEDGEMENT

This work was supported by ARTEMIS JU in the frame of the European FP7 research project Reconfigurable ROS-based Resilient Reasoning Robotic Cooperative Systems (R5COP), grant agreement no. 621447 and by the EU H2020 project UP-Drive under grant no. 688652.

REFERENCES

[1] J. Zbontar and Y. LeCun, "Computing the stereo matching cost with a convolutional neural network," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 1592–1599.

[2] W. Luo, A. G. Schwing, and R. Urtasun, "Efficient Deep Learning for Stereo Matching," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 5695–5703.

[3] H. Hirschmuller, "Stereo Processing by Semiglobal Matching and Mutual Information," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 2, pp. 328–341, Feb 2008.

[4] R. Spangenberg, T. Langner, S. Adfeldt, and R. Rojas, "Large scale Semi-Global Matching on the CPU," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, June 2014, pp. 195–201.

[5] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," in *Computer Vision ECCV '94*, ser. Lecture Notes in Computer Science, J.-O. Eklundh, Ed. Springer Berlin Heidelberg, 1994, vol. 801, pp. 151–158.

[6] D. Scharstein and R. Szeliski, "A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms," *International Journal of Computer Vision*, vol. 47, no. 1, p. 742, May 2002. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=64200

[7] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[8] M. Humenberger, T. Engelke, and W. Kubinger, "A census-based stereo vision algorithm using modified Semi-Global Matching and plane fitting to improve matching quality," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, June 2010, pp. 77–84.

[9] K. Yamaguchi, D. McAllester, and R. Urtasun, "Efficient joint segmentation, occlusion labeling, stereo and flow estimation," in *European Conference on Computer Vision*. Springer, 2014, pp. 756–771.

[10] F. Gney and A. Geiger, "Displets: Resolving stereo ambiguities using object knowledge," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 4165–4175.

[11] L. Schneider, M. Cordts, T. Rehfeld, D. Pfeiffer, M. Enzweiler, U. Franke, M. Pollefeys, and S. Roth, "Semantic Stixels: Depth is not enough," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, June 2016, pp. 110–117.

[12] S. Birchfield and C. Tomasi, "Depth discontinuities by pixel-to-pixel stereo," in *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, Jan 1998, pp. 1073–1080.

[13] R. Spangenberg, T. Langner, and R. Rojas, "Weighted Semi-Global Matching and Center-Symmetric Census Transform for Robust Driver Assistance."

[14] W. S. Fife and J. K. Archibald, "Improved Census Transforms for Resource-Optimized Stereo Vision," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 1, pp. 60–73, Jan 2013.

[15] M. Loghman and J. Kim, "SGM-based dense disparity estimation using adaptive Census transform," in *2013 International Conference on Connected Vehicles and Expo (ICCVE)*, Dec 2013, pp. 592–597.

[16] V. C. Miclea and S. Nedevschi, "Optimizing Census-based Semi Global Matching by genetic algorithms," in *2016 IEEE 12th International Conference on Intelligent Computer Communication and Processing (ICCP)*, Sept 2016, pp. 193–198.

[17] P. Pinggera, D. Pfeiffer, U. Franke, and R. Mester, "Know Your Limits: Accuracy of Long Range Stereoscopic Object Measurements in Practice," in *Computer Vision ECCV 2014*, ser. Lecture Notes in Computer Science, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Springer International Publishing, 2014, vol. 8690, pp. 96–111.

[18] Y. Cheng and L. Matthies, "Stereovision Bias Removal by Autocorrelation," in *Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on*, Jan 2015, pp. 1153–1160.

[19] I. Haller, C. Pantilie, T. Marita, and S. Nedevschi, "Statistical method for sub-pixel interpolation function estimation," in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, Sept 2010, pp. 1098–1103.

[20] I. Haller and S. Nedevschi, "Design of Interpolation Functions for Subpixel-Accuracy Stereo-Vision Systems," *Image Processing, IEEE Transactions on*, vol. 21, no. 2, pp. 889–898, Feb 2012.

[21] V.-C. Miclea, C.-C. Vancea, and S. Nedevschi, "New sub-pixel interpolation functions for accurate real-time stereo-matching algorithms," in *Intelligent Computer Communication and Processing (ICCP), 2015 IEEE International Conference on*, Sept 2015, pp. 173–178.

[22] C. C. Vancea, V. C. Miclea, and S. Nedevschi, "Improving stereo reconstruction by sub-pixel correction using histogram matching," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, June 2016, pp. 335–341.

[23] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[24] A. D. Costea and S. Nedevschi, "Fast traffic scene segmentation using multi-range features from multi-resolution filtered and spatial context channels," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, June 2016, pp. 328–334.

[25] G. Ghiasi and C. C. Fowlkes, "Laplacian Reconstruction and Refinement for Semantic Segmentation," *CoRR*, vol. abs/1605.02264, 2016. [Online]. Available: http://arxiv.org/abs/1605.02264

[26] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation," *CoRR*, vol. abs/1606.02147, 2016. [Online]. Available: http://arxiv.org/abs/1606.02147

[27] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 3431–3440.

[28] C. Banz, P. Pirsch, and H. Blume, "Evaluation of Penalty Functions for Semi-Global Matching Cost Aggregation," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 1–6, Jul. 2012.

[29] M. Menze and A. Geiger, "Object Scene Flow for Autonomous Vehicles," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[30] C. Pantilie and S. Nedevschi, "SORT-SGM: Subpixel Optimized Real-Time Semiglobal Matching for Intelligent Vehicles," *Vehicular Technology, IEEE Transactions on*, vol. 61, no. 3, pp. 1032–1042, March 2012.

[31] D. Scharstein, H. Hirschmuller, Y. Kitajima, G. Krathwohl, N. Nesic, X. Wang, and P. Westling, "High-Resolution Stereo Datasets with Subpixel-Accurate Ground Truth," in *Pattern Recognition - 36th German Conference, GCPR 2014, Munster, September 2-5*, pp. 31–42.

[32] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International Journal of Computer Vision*, vol. 111, no. 1, pp. 98–136, Jan. 2015.