Energy Minimization of Discrete Functions with Higher-order Potentials for Depth Map Generation

Dimitri Bulatov and Benedikt Kottler
Fraunhofer Institute of Optronics,
System Technologies and Image Exploitation
Gutleuthausstr. 1, 76275 Ettlingen, Germany
dimitri.bulatov@iosb.fraunhofer.de

Franz Rottensteiner
Institute of Photogrammetry und GeoInformation
Nienburger Str. 1, 30167 Hannover, Germany
rottensteiner@ipi.uni-hannover.de

***Abstract*—Minimization of discrete energy functions considering higher-order potentials is a challenging yet an important problem. In this work, a three-step procedure will be presented and exemplified on a general problem related to the dense depth map computation from multi-view configurations: Achieving a joint reconstruction of structure and semantics with piecewise planarity constraints. The three steps of the procedure are binarization, quadratization, and energy minimization. While the first and the third step are accomplished using procedures based on alpha-expansion and max-flow algorithms, respectively, we propose for the quadratization step a fast and simple module to reformulate the higher-order problem as a quadratic one. This module is based on edge statistics and is particularly useful for regular graphs and for third- or fourth-order potentials.**

I. Introduction and related work

Minimization of discrete energy functions is an indispensable tool for solving many problems in Computer Vision, an important example being the generation of depth maps from two or more images. Depth maps represent the most important intermediate result for 3D surface reconstruction from images and videos, because given an image with corresponding camera matrix, the 3D coordinates of the scene can be parameterized by means of scalar point depths. Depth map computation is usually based on a trade-off between a data-dependent photoconsistency term that can be computed for each pixel individually and a smoothness term that acts as a prior for the local configuration of depth values (e.g., assuming piecewise smoothness), which helps to propagate the depths of the salient pixels to regions of weak texture. In a Bayesian setting, the data term can be interpreted as the likelihood, whereas the smoothness term corresponds to the prior. This prior is conventionally modeled as the sum of deviations between depth values over *pairs of neighboring pixels*, and the problem can be described by a Markov Random Field (MRF), in which the image primitives (pixels) correspond to the nodes of an undirected graph while the photo-consistency term and the prior are referred to as unary and pairwise potentials, respectively. Examples are [1], which is a method specially developed for grid-based graphs, as well as [2], [3], which can be applied to arbitrary graphs.

However, as mentioned in [4], the pairwise smoothness terms (also called potentials) suffer from a number of problems, mostly because of being unable to "understand" the global connections between the data. For instance, if a scene contains many slanted (non-frontoparallel) or curvilinear surfaces, the output depth image may nevertheless contain many frontoparallel segments separated by depth jumps. The existence of numerous heuristic approaches to alleviate this problem demonstrates its actuality. For example, [5] suggested to decompose an image into triangles with vertices at points with known 3D coordinates; then, consideration of a triangle-based term helps to mitigate discretization artifacts and to support gradual changes of depth. However, only few contributions exist which worked with more general penalty terms in their optimization workflow. In [6], optimization is carried out over both, the disparity map itself and its gradients. At the cost of trebling the number of graph vertices, a pairwise smoothness term is applied. In the highly iterative optimization method, which depends a lot on the initial solution (normal vector field), no report on the minimized energy was presented. A piecewise planarity prior was formulated in [7], however, the resulting binarized graph was particularly tailored to the problem. Besides, no care was taken of the number of additional edges; it seems to depend on the number of images, even though depth is a scene property.

Another important way to handle textureless surfaces is to take a leap from pixels to higher-level instances. Thus, generalizing the smoothness terms for matching problems from pairwise to higher-order potentials has become increasingly popular in the recent years. We can mention simultaneous optimization of depth values and segmentation results [8], or even hierarchical sequences of (not necessarily depth) labels of pixels, segments and super-segments [4]. Finally, [9] and [10] extract the joint information about 3D structure and semantic properties of the scene using pairwise potentials and a relaxation approach, that is, subsequently fixing one kind of unknown variables (disparities or class) while optimizing over the other one. The joint reconstruction, however, presupposes simultaneous extraction of depth and context information. Here, higher-order potentials emerge as soon as classes *and* depths of *neighboring* pixels are considered to be correlated. These considerations are important because on the one hand reasoning on the semantics level can significantly improve the performance of scene reconstruction algorithms in occluded, weakly textured areas, areas of reflections, etc. On the other hand, the part of 3D geometry in recognition of objects is crucial especially if their appearances in the images are not discriminative enough. To our knowledge, optimization involving higher-order potential has been only successful for a rather sparse class of functions, like robust potentials [11].

Our goal is to propose a more general workflow for minimizing energy functions defined on 1) discrete random variables over 2) preferably regular graphs and 3) containing higher-order potentials. This workflow will be tested for a problem related to depth map estimation from images and reflecting contents of the previous paragraphs: Firstly, it will be a generalization of work of [7] which permits modeling the piecewise differentiability of the underlying surface. Secondly, given near-nadir aerial images, we simultaneously extract the information on 3D geometry and semantics; within the proposed joint minimization algorithm, the dependence of depth differences of two neighboring pixels on their class labels will be modeled by fourth-order potentials. The problem will be explained in Sec. II after a short reminder about image-based the data cost extraction for depth maps. The main contribution of the paper, Sec. III shows our algorithm for energy minimization. The three steps constituting this algorithm are *binarization*, which is slightly different to the conventional alpha-expansion, *quadratization* by means of a purpose-built modification of the algorithm of [12] and the method of [13], and *graph-based minimization* which worked well for both state-of-the-art methods [14] and [15]. The main goal of the paper is thus to demonstrate the feasibility of energy minimization with diverse higher-order potentials rather than sophisticated choice and adjustment these potentials. Thus, we compute the data cost for classification by means of a very simple algorithm and introduce heuristic mixed potentials of order four. The penalty terms for piecewise differentiability will be limited to the piecewise planarity and thus have the order three. We present the results for a well-known dataset in Sec. IV and summarize the contents of this article in Sec. V.

II. Problem statement

A. Preliminaries on multi-view cost aggregation

Given several images with known parameters of the underlying cameras, we wish to compute for every pixel $\mathbf{x}$ of a reference image J the depth $d_{\mathbf{x}}$, that is, the distance from the principal image plane to the first intersection of the camera ray emanating from this pixel with the object surface. The reference image may be one of the input images or an image at a virtual camera position. A hypothesized 3D intersection point can be projected into the input images, after which deviations based on a photo-consistency measure, $f(\cdot,\cdot)$, are obtained from pairs of the images $J_p(\mathbf{x}_{p,d})$ and $J_q(\mathbf{x}_{q,d})$ in a small neighborhood of projected points $\mathbf{x}_{p,d}$ and $\mathbf{x}_{p,d}$. To obtain the data energy term E_D, the values of f are aggregated to:

$$E_D(d_{\mathbf{x}}) = \sum_{p,q \in I} g_{p,q,\mathbf{x}} \cdot f(J_p(\mathbf{x}_{p,d}), J_q(\mathbf{x}_{q,d})), \tag{1}$$

where f is a photo-consistency term, such as normalized cross correlation, and I is a set of pairs of images to be considered. Finally, $g_{p,q,\mathbf{x}}$ are the aggregation weights which are important for modeling occlusions, because not all 3D points of the surface are visible in all images [5], [16]. In practice, this procedure can be significantly accelerated if the calculation of $\mathbf{x}_{p,d}$ is performed 1) without the intermediate result of 3D points and 2) simultaneously for a dense set of pixels. To omit calculation of 3D points, homographies induced by planes [17] are applied. The sweep-plane equation depends on the unknown $d_{\mathbf{x}}$ and the plane normal vector, which can either coincide with the normal of the principal plane of the reference camera or – in case of known absolute orientation – the horizontal plane. The latter approach has the advantage of less distortion of correlation windows in (1), see [6]. Once the homographies inducing planes have been computed, one can perform image-to-image warping and interpret all operations as filtering procedures, see [16] for details.

The energy E_D, computed using (1) at discretized values of d, is thus the first main input for our optimization problem:

$$L^* = \arg\min \left(E_D(L) + E_P(L,N) \right), \tag{2}$$

while the remaining terms L, N and E_P depend on the particular problem and will be explained in the following section.

B. Potentials for semantic depth maps

The random variables $\mathbf{x}_1, \ldots, \mathbf{x}_n \in J$ represent the n pixels of the image J and correspond to the nodes $\mathbf{X}$ of an undirected graph associated with J. Any possible assignment of labels to the random variables is called a labeling and denoted by L. Similarly to [9], we wish to compute both the 3D position (given by depth label d) and its semantic representation c of every pixel, thus leading to a *bivariate Markov Random Field*. For reasons of convenience, the graph is extended to a new graph G with $2n$ nodes. For the first n nodes, labels for depth are assigned while for the latter n nodes, the labels 1 to u of the variable c correspond to u classes, in our case $\{$*building, tree, grass, ground*$\}$, and $u = 4$. Overall, we have

$$L(\mathbf{x}) = (d_{\mathbf{x}}, c_{\mathbf{x}}) = (d(\mathbf{X}_d), c(\mathbf{X}_c)), \tag{3}$$

where $\mathbf{X}_d$ and $\mathbf{X}_c$ denote the same pixel in J and two different nodes in the graph G. The data term resulting from a labeling L in (2) is:

$$E_D(L) = \sum_{\mathbf{x}} E_D(d_{\mathbf{x}}) + \sum_{\mathbf{x}} E_D(c_{\mathbf{x}}), \tag{4}$$

where $E_D(d_{\mathbf{x}})$ is from (1). To compute $E_D(c_{\mathbf{x}})$, we first obtain features, such as relative elevation, normalized difference vegetation index, planarity and entropy. Finally, we combine these features into terms similar to those described by [18]. Note that the relative elevation is obtained from the preliminary result of depth map computation [5] followed by extraction of the digital terrain model (DTM) using [19].

We are now ready to describe the neighborhoods relations of the graph G as well as the corresponding smoothness priors E_P. We use the well-known notion of cliques, which are the fully-connected subsets of vertices of the graph. For example, a pairwise clique between the nodes $\mathbf{x}, \mathbf{y}$ implies that there is a (pairwise) penalty term involving their labels. There are three types of pairwise cliques in G. Firstly, for neighboring pixels in J (clique type $N_{2a} = \{\mathbf{x}, \mathbf{y} : \|\mathbf{x} - \mathbf{y}\| = 1\}$ in G), the differences of depths are punished by a truncated linear function:

$$E_P(d_{\mathbf{x}}, d_{\mathbf{y}}) = g_d(\mathbf{x}, \mathbf{y}) \cdot \min \left(|d_{\mathbf{x}} - d_{\mathbf{y}}|, m_d \right), \tag{5}$$

where $g_d(\mathbf{x}, \mathbf{y})$ is employed to represent radiometric similarity of $\mathbf{x}$ and $\mathbf{y}$: The more similar the colors, the higher g_d. However, in our first experiments, the terms m and g in this and upcoming potentials are constants.

Secondly, the term $E_P(c_{\mathbf{x}}, c_{\mathbf{y}})$ is the Potts function, see [9], Equation (2) with $g_c(\mathbf{x}, \mathbf{y})$) and the neighborhood in G is denoted by $N_{2b} = N_{2a}$). Thirdly, joint potentials, defined for the

neighborhood type $N_{2c} = \{\mathbf{X}_d, \mathbf{X}_c\}$, are supposed to monitor how the depth of the pixel $\mathbf{x}$ influences its class:

$$E_P(d_\mathbf{x}, c_\mathbf{x}) = \gamma_{cd}\left(c, \tilde{d}(\mathbf{x})\right). \quad (6)$$

Here, $\tilde{d}$ is the relative elevation, which for an almost plain terrain can be set independent on $\mathbf{x}$, and $\gamma_{cd}(\cdot, d)$ is a sigmoid function, which is monotonically increasing if c corresponds to one of both classes *grass* or *road* and is decreasing for the *building* and *tree* class.

In order to allow the piecewise planarity of the surface, we introduce the third-order potentials defined over cliques $N_3 = \{\mathbf{x}-\mathbf{v}, \mathbf{x}, \mathbf{x}+\mathbf{v}\}$, where $\mathbf{x} = (x, y)$ and $\mathbf{v}$ takes on one of two values (0, 1) or (1, 0), except of pixels at the image border:

$$E_P(d_\mathbf{x}, d_{\mathbf{x}\pm\mathbf{v}}) = g_{\mathbf{x},\mathbf{x}\pm\mathbf{v}} \cdot \min\left(|d_{\mathbf{x}-\mathbf{v}} - 2d_\mathbf{x} + d_{\mathbf{x}+\mathbf{v}}|, m_{\mathbf{x},\mathbf{x}\pm\mathbf{v}}\right). \quad (7)$$

Finally, we define four-order cliques $N_4 = \{\mathbf{X}_d, \mathbf{X}_c, \mathbf{Y}_d, \mathbf{Y}_c\}$ for all $(\mathbf{x}, \mathbf{y}) \in N_{2a}$ and the corresponding fourth-order potential:

$$E_P(d_\mathbf{x}, d_\mathbf{y}, c_\mathbf{x}, c_\mathbf{y}) = m \cdot \min\left(|d_\mathbf{x} - d_\mathbf{y}|, \gamma_{cc}(c_\mathbf{x}, c_\mathbf{y})\right), \quad (8)$$

where $\gamma_{cc}(c_\mathbf{x}, c_\mathbf{y})$ is 0 if $c_\mathbf{x} \neq c_\mathbf{y}$. In case $c_\mathbf{x} = c_\mathbf{y} = \mathit{tree}$, $\gamma_{cc} = 0.5$ and otherwise, it is 1. This is done to permit more variance in depth of neighboring pixels if they belong to the tree class or to different classes. Combining equations (5)-(8), we have:

$$E_P(L, N) = \sum_{N_{2a}} E_P(d_\mathbf{x}, d_\mathbf{y}) + \sum_{N_{2b}} E_P(c_\mathbf{x}, c_\mathbf{y}) \quad (9)$$

$$+ \sum_{N_{2c}} E_P(d_\mathbf{x}, c_\mathbf{x}) + \sum_{N_{3c}} E_P(d_\mathbf{x}, d_{\mathbf{x}\pm\mathbf{v}}) + \sum_{N_4} E_P(d_\mathbf{x}, d_\mathbf{y}, c_\mathbf{x}, c_\mathbf{y}).$$

We emphasize that further modifications are possible; for example, a meaningful combination of (7) and (8) for the class *building* would yield six-order potentials. We also note that the functional (9) is the generalization of both problems [7] and [9]. The energy function (9) without N_3 neighborhoods was minimized by [9] by subsequently keeping one of the variables c and d constant while optimizing over the other ones, thus neutralizing the effect of the fourth-order potentials. In Sec. III, we will show how to perform a simultaneous energy minimization over all $2n$ nodes of G. This leads to higher computational effort of $O(|c| \cdot |d|)$ instead of $O(|c| + |d|)$ for non-simultaneous approaches, but allows a wider range of moves and thus a better local minimum.

III. MAIN ALGORITHM

A. Binarization

The quintessence of a move-making algorithm is to represent the energy function as a problem in binary variables. Consider a mapping B from $\mathbf{X}$ to $\{0, 1\}$. For example, in the case of alpha-expansion, given an initial configuration L_0 and $i \in \mathbf{X}$, we have: $B(i) = (b_i) = 0$ if $L(i) = L_0(i)$ and $b_i = 1$ if $L(i) = \alpha$. As it was pointed out in [7], B may serve as a switch between two configurations L_0 and L_α. This observation is particularly important for our problem because we define the configuration $L_1 = L_\alpha$ to take on fixed values of d for the first n nodes and fixed values of c for the latter n nodes. Thus, the optimization runs over all pairs (d, c), possibly in a randomized order, while the local solution $L_0 = \arg\min E_D(L)$ serves as the initialization and it is subsequently updated after each iteration.

For binarization, we define the binary potentials U_i^0, U_i^0 (unary), $U_{ij}^{00}, U_{ij}^{01}, U_{ij}^{10}, U_{ij}^{11}$ (pairwise), $U_{ijk}^{000}, ..., U_{ijk}^{111}$ (triple), etc., such that the energy of the problem formulated in binary variables is the same as that of the corresponding configuration. That is, for each node i and for each clique $\{i_1, ..., i_n\}$, we have:

$$U_i^{b_i} = E_D(L_0(i)) \text{ and } U_{i_1,...,i_n}^{b_1,...,b_n} = E_P\left(L_{b_1}(i_1), ..., L_{b_n}(i_n)\right), \quad (10)$$

where $L(i)$ as in (3). Using libraries optimized for matrix operations, the terms U can be computed very fast for different choices of functions E_D, E_P. The goal of the next section is to eliminate the higher-order potentials in (10), because the vast majority of energy minimization algorithms applicable in practice handle functionals with second-order potentials only.

B. Quadratization

The common way to get rid of higher-order potentials is to calculate coefficients of a so-called Pseudo-Boolean (PB) polynomial function

$$E(B) = a_0 + \sum_i a_i b_i + \sum_{N_2} a_{ij} b_i b_j + \sum_{N_3} a_{ijk} b_i b_j b_k + \cdots, \quad (11)$$

and then to replace it by another one which has the same minimum as the original problem, but is of at most second degree. The PB polynomial (11) is binary in the variables b_i and has the same degree as the highest potential order. The calculation of its coefficients a_0, a_i, etc. from the potentials in (10) is carried out using the recursive algorithm presented in [12]. Note that in general, the neighborhoods $N_2, N_3, ...$ etc. may change because sub-neighborhoods from higher-order potentials add to lower-order neighborhoods. Collecting terms from the additional higher-order sub-neighborhoods is not trivial for big problems unless considerable memory is provided in order to keep all the occurring combinations of variables. At present, we use sparse matrices formats for second-order neighborhoods, while for higher orders, union find algorithms as well as special data types (associative containers) could be used. However, in our special case it is easy to see that thanks to the regular graph structure, a higher-order term as in (11) can stem from exactly one higher-order potential $U_{i_1,...,i_n}^{b_1,...,b_n}$, where $n = 3$ or 4.

Starting from the function given in (11), our intention is to obtain a modified energy function consisting of at most quadratic terms, but having the same minimum as the original problem. This can be achieved by replacing some products of variables by new, so-called *dummy* variables. In the following, we will describe our modification of one of the first reduction methods, which was adopted and didactically well-prepared by [12], and the state-of-the-art method of [13].

The key idea of the algorithm described in [12] is to replace any occurrence of the product $b_i b_j$ by β and add $M \cdot (b_i b_j + 3\beta - 2b_i\beta - 2b_j\beta)$ to the energy function $E(B)$. The added term links the dummy variable in the way that the condition $\beta = b_i b_j$ cannot hold unless B is the global or a strong local minimum of (11). This can be guaranteed for $M > -A$, where A is the sum of all negative coefficients. The remaining question is the strategy of choice of products to minimize the number of dummy variables, which is itself an NP-hard problem. However, we use the fact that the maximal clique size is moderate in order to obtain a fast procedure

based on the edge statistics (ES) which aims at reducing the number of dummy variables.

During collection of polynomial coefficients of the second order into sparse matrices in step 1, we obtain information about the occurrence of edges. Then, within an inner loop over the cliques of highest order, for the current clique, we find an edge which either was already marked as a dummy variable before or has the highest occurrences statistics. In the latter case, this edge is marked as a dummy variable as well and replaced in all subsequent occurrences. After this inner loop, the maximal clique size is reduced by one and the algorithm begins again, until at most second-order terms remain.

The main disadvantage of the algorithm of [12] and the modification ES is the presence of many positive coefficients resulting from the terms of the type $M \cdot b_i b_j$ in the reduced polynomial. Such terms represent non-submodular edges of the auxiliary graph constructed for the minimal cut computation; however, in such graphs, global minimum cut extraction cannot be performed in polynomial time. Moreover, the computational complexity is directly connected to the number of submodular edges. Therefore, the algorithm of [13], which will be denoted by *Fix* throughout this paper, aims at the elimination of as many non-submodular edges as possible. In the first step, the positive higher-order terms are replaced by linear combinations of terms that either are positive and have a smaller degree, or that are negative and have at most the same degree; in this step, the dummy variables replace some common subset of original variables. As soon as there are no higher-order positive terms, the higher-order negative terms are replaced by linear and submodular quadratic terms.

From the coefficients of the reduced polynomial (like in (11), but with at most quadratic terms $a_{ij}b_i b_j$), one can easily obtain proper potentials for the upcoming energy minimization. For example, we choose the function:

$$\tilde{E}(B) = \sum_{i \in \tilde{N}_1} V_i^{b_i} + \sum_{(i,j) \in \tilde{N}_2} V_{ij}^{b_i, b_j} \text{ with} \tag{12}$$

$$V_0^i = a_0/|\tilde{N}_2|, V_1^i = a_i - a_0/|\tilde{N}_2|, V_{ij}^{00} = V_{ij}^{01} = V_{ij}^{10} = 0, V_{ij}^{11} = a_{ij},$$

which is equivalent to (11) except for the additive constant a_0.

C. Energy minimization

The task of this section is to obtain a strong local minimum for the energy functional given by the unary and pairwise potentials V as formulated in (12). As the energy functional is highly non-submodular (not even in its original formulation!) its global minimum cannot be obtained in polynomial time [15]. Introducing dummy nodes with edge weights having extremely large positive values a_{ij} in (12) bears the risk of ending up in a *weak* local minimum, when the energy of the obtained configuration exceeds that of the initial configuration, because one or several dummy variables take on a forbidden value $\beta \neq b_i b_j$. Nevertheless, we identified two methods which always yield configurations of non-increasing energy.

First, we considered the Boykov-Kolmogorov (BK) algorithm based on [14]. In the graph constructed for computation of maximum flow, the edges with negative capacities are ignored every time while searching for the augmented path. Thus, the resulting energy $E(B_1)$ does not exceed the previous energy $E(B_0)$, though B_1 does not necessarily achieve the global minimum of (12) when there are no remaining augmenting paths. This method has turned out to be significantly faster that others for the problem of Sec. II.

Second, we employed the modified Quadratic Pseudo-Boolean Optimization (QPBO) algorithm [15]. In this method, called QPBO+I (Improve), a subset of vertices stemming from an approximating configuration B_0 is fixed and the original QPBO algorithm [20] is run to obtain the labeling B_1 for the remaining nodes. The algorithm in its original implementation leaves some nodes unlabeled if there are non-submodular terms; hence, replacing these nodes by values from B_1 yields a configuration which does not have unlabeled nodes anymore and for which, as has been proved in [15], $E(B_1) \leq E(B_0)$. Now, the algorithm is run again, but with solution B_1 and a different subset of fixed nodes. Such an iterative application of the QPBO+I algorithm leads to a considerable reduction of E, however, at the cost of computation time.

IV. Experimental results

The dataset used for our experiments is the well-known Vaihingen benchmark dataset, which was generated by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF) [21]. Several images with corresponding camera matrices are provided. We applied [16] with changes explained in Sec. II to five of these images in order to compute the data energy. The reference image and the local result are shown in Fig. 1. To evaluate the performance of our algorithm, we consider two subproblems. The first problem, denoted by P1, is depth map extraction using piecewise planarity term. This term makes sense as long as the number K of labels for depth (in our case, it is elevation) is large. Indeed, we are interested in finding parallaxes at subpixel level. It is also necessary to perform *linear* discretization of elevation values while calculating data energy. That is, $d_k = (k d_{\max} + (K-k) d_{\min})/K$, where $k = 0, ..., K$. For the second problem (P2), namely joint reconstruction of scene and semantics, K should not be too high, since otherwise the term (8) will tend to compute trees in all regions with changes of elevations. Thus, the term (7) was dropped. In addition, because of a quite high number of submodular edges, sub-images must be considered in order to obtain tractable computation times. For the results reported for the problem P2 in Table I, we worked with 300×400 images as well as 20 depth and 4 class labels. From all methods we mentioned, only BK could handle the image with 700×500 pixels and 20 labels in a reasonable time: 15 s per iteration and below one hour in total.

We illustrate in the graphs on the left of Fig. 2 and Fig. 3 as well as in Table I the performance on the energy minimization and computing times for several combinations of the quadratization (ES and Fix from [13]) and energy optimization (BK) methods. For example, the combination ES+BK+QPBO+I^n means that the proposed method on quadratization is followed by the BK optimization, and then by n applications of the Improve method. Other methods we tested did not result in a decrease of the energy; even in the case where non-submodular terms were omitted, the decrease of energy was slower than of the methods discussed above. It becomes clear that the BK algorithm achieved by far the fastest performance without shortcomings in the quality. However, the global minimum

was not reached, since by applying QPBO+I to the result, we obtain a further energy decrease. There were two fast combinations ES+BK and Fix+QBPO, for which the time per iteration is about 0.21 and 0.36 seconds, respectively. While we could not yet test the combination Fix+BK, it is not recommendable to combine ES and QPBO(+I), because the ES method yields many submodular edges: Applying QPBO+I brings about very high computing times while without I, too many nodes remain unlabeled and the energy decrease is thus less significant. Still, submodular edges also remain in quadratization result of [13], because otherwise we would have achieved the global minimum. Also, we see that the local result is already rather good such that the reduction of energy is only around 1.3-1.6%. With respect to the performance of the quadratization method alone, we observed that for the problem P1 the original implementation of [12] and its modification ES yielded, averagely per iteration, $4.1 \cdot 10^4$ and $2.4 \cdot 10^4$ extra variables, respectively, while the quadratization using [13] yielded $1.23 \cdot 10^5$ dummy variables. Thus, the numbers – together with a lower computing times achieved after BK minimization – speak for our method.

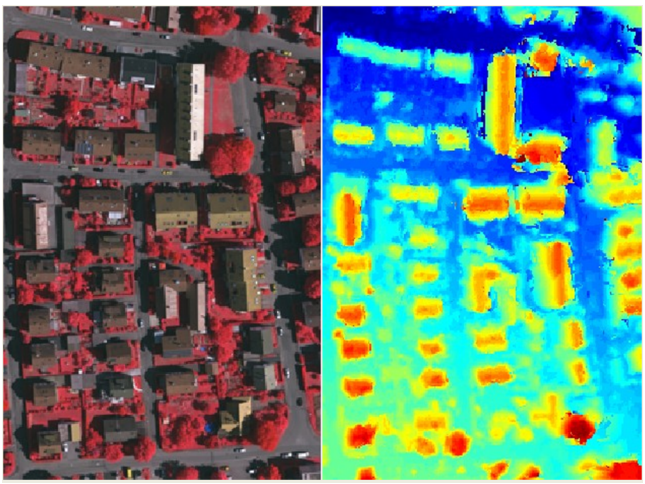

Fig. 1. Left: Reference image for the considered dataset. Right: result of the per-pixel minimization of E_D

Turning our attention to qualitative results, one can note in the height profiles of Fig. 2, middle, the linear change of depth for the green curves, where piecewise planarity is better visible than in the noisy local result (red) and the optimization with only pairwise smoothness term, which is susceptible to fronto-parallel planes (blue). Especially in the areas where part of the wall is visible, the gradual change of depth can be observed. With respect to the joint reconstruction, shown in Fig. 3, middle and right, we see in sub-image (A) the rather noisy result for the depth reconstruction using only data cost. We see that an application of the semi-global optimization [1] followed by median filtering (C) improves the elevation maps, but the building outlines are not that much visible as in the result (E) of the joint optimization. Starting from (C), the result of classification using merely the four features is shown in image (B). Since no smoothness prior is used, the features for neighboring pixels do not "know" anything about each other and the result appears noisy. Running semi-global optimization using likelihoods extracted from the features slightly improves the output (D); however, the information about elevation is considered only implicitly. This could also be the danger, though to a smaller extent, in case of relaxation as in [9]. Only in the proposed method of joint reconstruction, the elevations are consistently included into optimization; therefore the classes in image (F) look more plausible, especially in the shadow region between trees in the top right corner, see reference image (G). Unfortunately, the intuitive, heuristic choice of the smoothness parameters and terms does not allow a further, more successful improvement of the joint reconstruction result. The parameters g_d, g_c, γ_{cd} and γ_{cc} from Sec. II-B are empirically chosen and their combination is certainly not optimal. Furthermore, the introduced terms do not always represent the actual model of the ground-truth. For instance, the mixed unary potentials do not make much sense without the DTM, which for itself is an output of the interpolated elevation map. Also, elevation is basically the only feature which differentiates between buildings and ground, as well as trees and grass. The undesired consequence of this is that the data cost features and the smoothness priors become highly correlated with each other. However, this work demonstrates the proof of concept that higher-order energy functions, which bear enormous potential for problems related with dense reconstruction, can be efficiently minimized.

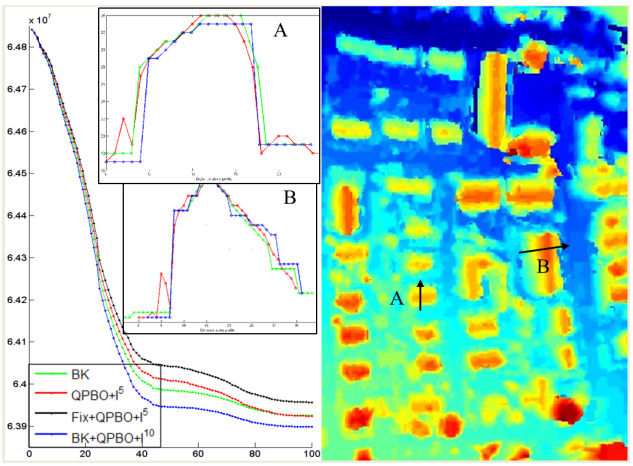

Fig. 2. Left: Energy minimization, performance of different methods for problem P1. Middle: Height profiles obtained from two black line segments on the right, with directions left-to-right specified by arrows. The red lines are from the local result (shown in Fig. 1, right). The results of non-local optimization method with only second-order and only third-order potentials are shown by blue and green curves, respectively. Right: Result of the non-local optimization.

V. Conclusions and Outlook

We presented a method for energy minimization which is fully automatic and modular, that is, each of the three modules (binarization, quadratization, energy minimization) can be modified or replaced as soon as a better version is available. Currently, the optimization method [14] achieved the best performance, but probably, it can be accelerated or improved in the future with the quadratization result of [13]. This method aims at the elimination of non-submodular terms, contrary to our procedure, which is based on edge statistics and strives for reducing the number of extra variables. We achieved a reduction to up to 41% and 80% compared, respectively, to the original method and that of [13].

TABLE I. RUNNING TIME FOR AND DECREASE OF ENERGY (IN %) AFTER ONE EXPANSION CYCLE (OVER ALL LABELS).

method	ES+BK	ES + QPBO+I^5	Fix + QPBO+I^5	ES+BK+ QPBO+I^{10}
P1, time, s	17	745	76	1404
P1, energy%	1.32	1.23	1.37	1.38
P2, time, s	88	5510	178	10911
P2, energy, %	1.50	1.47	1.49	1.50

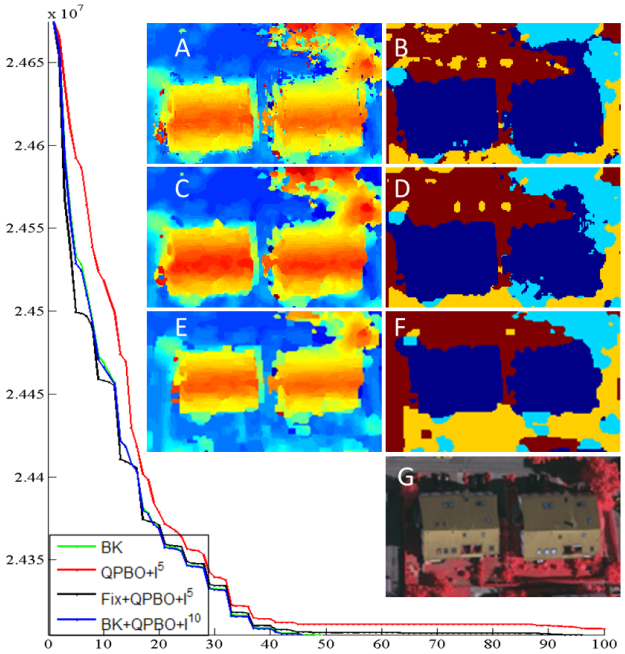

Fig. 3. Left: Energy minimization, performance of different methods for problem P2. Middle: results for the elevation computation. Right: Classification with *building*, *tree*, *grass* and *street* classes marked by blue, light blue, orange and red, respectively. Bottom right: orthophoto fragment. See text for more comments.

The established workflow for energy minimization can be applied to other problems involving higher-order potentials, such as extraction of active contours or fields-of-experts for image restoration. Here this method was exemplified for a challenging problem related to simultaneous reconstruction of dense 3D structure and semantics. The considered problem is a generalization of two previous works [7] and [9]. In our future work, we have to perform quantitative evaluation of the deviation of the elevation map and the classification result from the reference. To achieve good results, it will be important to modify the heuristic potentials of Sec. II-B for $E_D(c)$. They and also the smoothness parameters should ideally be subject of a training procedure, which, additionally, can enable consideration of more classes than only four. The advantage of the proposed method is that once the potentials of (5)-(8) are modified, the computing time is not significantly changed. For further reduction of computation time, four propositions can be taken into consideration: better exploitation of information about underlying graphs, using segments (super-pixels) instead of pixels, discarding "hopeless" combinations of labels for elevation and class during alpha-expansion, and a decomposition of the input image into overlapping tiles.

REFERENCES

[1] H. Hirschmüller, "Stereo processing by semi-global matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, 2008.

[2] V. Kolmogorov, "Convergent tree-reweighted message passing for energy minimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1568–1583, 2006.

[3] J. Sun, N.-N. Zheng, and H.-Y. Shum, "Stereo matching using belief propagation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 787–800, 2003.

[4] L. Ladicky, C. Russell, P. Kohli, and P. Torr, "Associative hierarchical random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 6, pp. 1056–1077, 2014.

[5] D. Bulatov, P. Wernerus, and C. Heipke, "Multi-view dense matching supported by triangular meshes," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 6, pp. 907–918, 2011.

[6] G. Li and S. W. Zucker, "Differential geometric consistency extends stereo to curved surfaces," in *Computer Vision–ECCV 2006*. Springer, 2006, pp. 44–57.

[7] O. Woodford, P. Torr, I. Reid, and A. Fitzgibbon, "Global stereo reconstruction under second-order smoothness priors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2115–2128, 2009.

[8] M. Bleyer, C. Rother, P. Kohli, D. Scharstein, and S. Sinha, "Object stereojoint stereo matching and object segmentation," in *IEEE Conf. on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 3081–3088.

[9] L. Ladicky, P. Sturgess, C. Russell, S. Sengupta, Y. Bastanlar, W. Clocksin, and P. Torr, "Joint optimization for object class segmentation and dense stereo reconstruction," *International Journal of Computer Vision*, vol. 100, no. 2, pp. 122–133, 2012.

[10] M. Menze and A. Geiger, "Object scene flow for autonomous vehicles," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2015, pp. 3061–3070.

[11] P. Kohli, L. Ladický, and P. H. S. Torr, "Robust higher order potentials for enforcing label consistency," *International Journal of Computer Vision*, vol. 82, no. 3, pp. 302–324, 2009.

[12] A. M. Ali, A. A. Farag, and G. L. GimelFarb, "Optimizing binary MRFs with higher order cliques," in *Proc. European Conference on Computer Vision*. Springer, 2008, pp. 98–111.

[13] A. Fix, A. Gruber, E. Boros, and R. Zabih, "A graph cut algorithm for higher-order markov random fields," in *IEEE Intern. Conf. on Computer Vision*, 2011, pp. 1020–1027.

[14] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, 2004.

[15] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer, "Optimizing binary MRFs via extended roof duality," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[16] D. Bulatov, "Temporal selection of images for a fast algorithm for depth-map extraction in multi-baseline configurations," in *Intern. Conf. on Computer Vision Theory and Applicationsn*, 2015, pp. 395–402.

[17] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, 2000.

[18] F. Lafarge and C. Mallet, "Creating large-scale city models from 3D-point clouds: a robust approach with hybrid representation," *International Journal of Computer Vision*, vol. 99, no. 1, pp. 69–85, 2012.

[19] D. Bulatov, P. Wernerus, and H. Gross, "On applications of sequential multi-view dense reconstruction from aerial images," in *Proc. of the 1st International Conference on Pattern Recognition Applications and Methods*, 2012, pp. 275–280.

[20] E. Boros, P. L. Hammer, and G. Tavares, "Preprocessing of unconstrained quadratic binary optimization," *Technical report RRR 10-2006*, 2006.

[21] M. Cramer, "The DGPF test on digital aerial camera evaluation – overview and test design," *Photogrammetrie, Fernerkundung, Geoinformation*, vol. 2, pp. 73–82, 2010.