Alignment of Building Footprints Using Quasi-Nadir Aerial Photography

Dimitri Bulatov[1,2](✉)

[1] Department Scene Analysis, Fraunhofer IOSB,
Gutleuthausstr. 1, 76275 Ettlingen, Germany
dimitri.bulatov@iosb.fraunhofer.de

[2] Department of Spatial Sciences, Curtin University of Technology,
Perth, WA 6102, Australia
http://www.iosb.fraunhofer.de

Abstract. In this paper, we consider the alignment problem of building outlines, provided by openly available sources, and high resolution aerial images. This problem can be transferred to that of matching images with different modalities. After studying related works, we propose to minimize a cost function penalizing both color and gradient discrepancies. Semantic context is extensively taken into account, and additional information, such as classification result, can be integrated. Pyramid-based coarse registration and median-filtering-based outlier suppression were implemented as pre- and post-processing modules, respectively. We performed extensive tests with three very different datasets and achieved encouraging results, which were very stable once application of pre- and post-processing took place.

Keywords: Alignment · Building outlines · GIS · Multimodal registration · Optimization

1 Introduction and Previous Research

Motivation. Buildings represent essential part of urban infrastructure and their correct geo-localization is important for many applications, such as city planning, civil security, and disaster management. In this last application, up-to-date information about building footprints are needed both by emergency services for setting up rescue missions, and insurance companies, who in a shortest possible time must assess the damage and restitute to the policyholder the relevant amount of money. The company usually sends a so-called loss adjuster to assess the damage of each building in the company's portfolio. To spare this loss adjuster the tedious process of roof-climbing, close-range quasi-nadir aerial or UAV images with a very high resolution are increasingly being applied to assess the roof damages. Automatic evaluation of damage by means of image processing methods, such as those described by [9] and [15], is another quite efficient concept of saving costs for the insurer. A very important detail about

M. Felsberg et al. (Eds.): SCIA 2019, LNCS 11482, pp. 361–373, 2019.
https://doi.org/10.1007/978-3-030-20205-7_30

both contributions is that damage assessment is carried out not building-wise but region-wise, which allows to assess whether tiles were blown away by the wind, damaged by heavy trees, or whether the roof was entirely collapsed. From this information, the damage degree and thus the compensation amount can be calculated. However, the necessary assumption for this is a correct delineation of buildings in the portfolio. Bearing this application in mind, the task covered in this paper will be alignment of building footprints provided by GIS data with the relevant image data. As [4,16] have pointed out, the deviations between the GIS database and the image can be quite high (≈8 m) and the main reasons for this are: the three-dimensional character of buildings, occasional discrepancy between roof polygons and ground plans, as well as changes with respect to out-to-date database. We wish to exclude these coarse systematic errors and accomplish the alignment task with the high-resolution image data and available geographical data only, keeping in mind that time is a critical factor and that training examples are hardly available or useful because some buildings may be damaged or destroyed. In what follows, we will briefly review the existing approaches, explain their insufficiencies, and outline our contributions.

Previous Works. We start with the interactive approach [2], where images were segmented with a commercial software, after which segments were processed manually. Since for large scenes, interactive processing is cumbersome and since we wish to make use of available outlines, we turned our attention towards automatic methods. Active contour models [10,11], such as snakes, are helpful for evolution of already available approximate values, but are often an overkill if alignment transformations can be described by a few parameters, to the same extent as methods based on fitting preferably rectangular primitives, as do [1] by means of Marked Point Processes. For rigid alignment transformation, matching key-points, such as [8], and model instantiation using RANSAC is probably the fastest strategy. Here two main challenges are incorporating the context information (typical properties of buildings) and matching key-points in extremely different images. Furthermore, there exist change detection methods, such as [3], where a subdivision into new, modified, remaining, and not-anymore-existing building was presented. Particularly interesting is their suggestion to take local maximums of gradient maps as seed points for building outlines. However, 3D data must be acquired from the satellite images. As for purely image-based methods, convolutional neural networks (CNNs) are increasingly being applied for outlining [10,13] and aligning [14,16] buildings. The latter contribution establishes analogy with the traditional gradient descent method while the conventional matching based on pyramids inspired the design of their neural network architecture (encoder-decoder like ones or with U-connections as in [12]). The big advantage of a CNN-based approach is that it can easily be generalized to other problem settings, such as medical image registration, however, we find it a pity that the well-known properties of buildings were sacrificed in favor of dozens of training examples. Finally, [14] propose a rather flat architecture that allows them to obtain a *heatmap* (of inverse likelihood) as data cost term.

Optimization runs over all offsets, as random variables on a Markov Random Field, using this data term as well as the usual smoothness assumption that neighboring buildings must have similar offsets. In our work, we will avoid computing heatmaps since for high-resolution data (building masks having 40000 and more pixels as well as 60 pixels offset), this could be costly. Thus, the aforementioned non-local energy minimization is simplified to a procedure reminding median filtering.

Most interesting to the authors are procedures operating without training data and possibly independent on resolution. Variational approaches are excellent examples for this. In [5], intensity values are interpreted as samples of two random processes and are linked by a probability density function (such as Mutual Information). The transformation is computed pixelwise and a regularization term is added to penalize transformations of neighboring pixels. What we consider as a bottleneck is the gradient-based energy minimization scheme because, especially for destroyed buildings, outliers must be treated with care. At cost of computation time, we will minimize a median-based energy function by means of the downhill simplex algorithm.

Contributions. We interpreted the alignment problem between the rasterized building outline and the image fragment as registration of two multimodal images and implemented a robust and fast approach for obtaining the unknown registration parameters. Technically, an energy function consisting of a color consistency and a gradient consistency terms is minimized. These terms are specific for the current building, but they are weighted by factors taking general building properties into account. In spite of an occasional presence of destroyed buildings, neither training data is required nor retrieving pixelwise cost functions (heatmaps). Even though it does not fully apply to the pre- and post-processing modules, the core part of the approach shows a very similar behavior for datasets having a different resolution. Finally, a classification result can be easily considered yielding better results and making dispensable the gradient-based term.

2 Methodology

The main part of our work is organized as follows. The most common variable names and basic definitions will be provided in Sect. 2.1, after which we give the cost function and details on its minimization (Sect. 2.2) concluded by the pre- and post-processing modules (Sect. 2.3).

2.1 Preliminaries

For a large amount of cities, there exist GIS data for building footprints ($\mathcal{P}$) that we wish to align with the actual image data. Let $\mathcal{I}$ denote a region of interest in a geo-referenced airborne image with three or more channels, containing a building with some surrounding area and let $\mathcal{M}$ be the mask obtained by rasterization of the corresponding footprint $\mathcal{P}$. That is, for points inside of the polygon, the value of $\mathcal{M}$ is 1 and outside it is zero (see Fig. 2 of [16]). A special value (2) is given to pixels at the border of the rasterization. We are looking for a transformation φ to align $\mathcal{M}$ with the roof of the building that way that the corresponding edges of the transformed outline $\mathcal{P}(\varphi)$ coincide with the roof silhouettes in $\mathcal{I}$. For φ, we consider a two-dimensional translation within a known range (search range), but if necessary, our approach can be generalized for a four- or even six-dimensional vector representing an Euclidean or affine transformation, respectively.

2.2 Minimization of Energy Function

In the case that freely available building outlines and roofs fit quite well, there is a sufficient overlap between the building footprint and the requested roof area to guarantee good starting values for our target function. Consequently, a modification of the mutual information can be applied, taking into account the homogeneity of the dominant color $\mathbf{f} \in \mathbb{R}^3$ sampled from a 3D histogram over the color values of all pixels $\mathbf{p}$ in $\mathcal{I}$ labeled as *inside* of $\mathcal{M}$. The number b of the histogram bins was 16; that is, color resolution of 16 for an 8-bit image. Sometimes, a building roof contains more than one dominant color, which is mostly true if the building is destroyed. Additionally, there may be some considerable fair or dark spots, like dormers, chimneys, or their shadows. To cope with this, the penalization between $\mathcal{I}$ and the dominant color $\mathbf{f}$ over the channels of and later over pixels is carried out using the L_1-norm which is more robust to such outliers. We denote this penalization (our *first* energy term) by $\|\mathcal{I}_{\mathbf{f}}(\mathbf{p})\|$. An additional weighting $w_{\mathbf{f}}$ can be applied according to how likely a pixel is supposed to belong to a building. Ideally, this should reflect the likelihood for the building class in the classification result $\mathcal{C}(p)$. In order not to lose too many resources for classification, we consider merely the pixelwise NDVI (Normalized Difference Vegetation Index) measure rescaled between 0 and 1. This measure is very popular in Remote Sensing if it comes to separate buildings from vegetation. We applied the term $\mathcal{C} = (1 + \mathcal{R}/\mathcal{N})^{-1}$, which is close to 0 if the near infra-red channel $\mathcal{N}$ is negligible compared to the red channel $\mathcal{R}$ while in the opposite case, it is 1.

As our *second* energy term, we wish to enforce the norm of the image gradient to be significantly higher at *border* pixels than *inside* $\mathcal{P}(\varphi)$. Analogously to $w_{\mathbf{f}}$, the weighting w_{∇} takes on the minimum value on the border, a small positive value inside and, as an option, a smoothly decreasing function outside the mask, since around buildings, high texture variations (gardens, roads, cars) are often present. The overall cost function is thus

$$E(\varphi) = \sum_{\mathbf{p} \in \mathcal{M}(\varphi)} \left\{ \alpha w_{\mathbf{f}}(\mathbf{p}) \|\mathcal{I}_{\mathbf{f}}(\mathbf{p})\| + (1-\alpha) \tilde{w}_{\nabla}(\mathbf{p}) \|\nabla \mathcal{I}(\mathbf{p})\| \right\}, \tag{1}$$

where

$$w_{\mathbf{f}}(\mathbf{p}) = \begin{cases} \mathcal{C}(\mathbf{p}) \text{ OR } 1 & \text{if } m > 0, \\ 0 & \text{if } m = 0 \end{cases}, \quad w_{\nabla}(\mathbf{p}) = \begin{cases} -1 & \text{if } m = 2, \\ 0.01 & \text{if } m = 1 \\ -e^{\frac{-d(\mathbf{p})}{\sigma}} \text{ OR } 0 & \text{if } m = 0 \end{cases}, \tag{2}$$

and $m = \mathcal{M}(\mathbf{p})$. Furthermore, d is the distance from $\mathbf{p}$ to where $\mathcal{M}$ is 1 (to be computed as a morphological operation at the binary image patch), σ is a constant around 0.5, α is a balance parameter to be explored in the experiments section together with both OR options in (2), and $\tilde{\cdot}$ denotes Gaussian smoothing. Note that while both w.-terms in (1) are supposed to fit the outline $\mathcal{P}$ to *a* building, both terms involving $\|\cdot\|$ pull $\mathcal{P}$ to *the* relevant building.

To minimize (1), we used the gradient-free Nelder-Mead method implemented by [7]. Its big disadvantage is to get occasionally stuck in a local minimum. However, it is several orders of magnitude faster than simulated annealing. Out of this reason, the Nelder-Mead method has been run for several starting values of φ after which the value yielding the minimum energy is chosen. At a lower resolution, an alternative to this method is to perform the exhaustive search for every single integer offset. Even though it is neither feasible for high resolution images nor for higher-dimensional search space, computation of cost function can be performed as a sliding window approach, with a sequence of convolutional operators (similar to CNNs), thus allowing to obtain a heatmap $\mathcal{H}(\varphi)$, see Fig. 1.

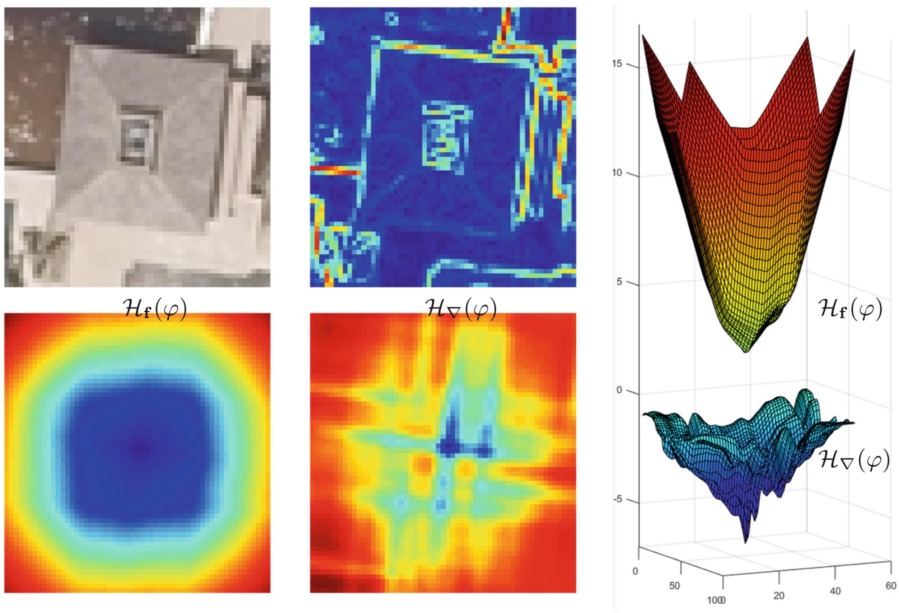

Fig. 1. Visualization of input data and cost function: top left and middle: $\mathcal{I}$ and $\nabla\mathcal{I}$. Bottom left and middle: Heatmaps $\mathcal{H}$ induced by both terms in (1), whereby blue means low energy/high likelihood. Right: heatmaps $\mathcal{H}$ represented as 3D surfaces. Note the numerous side minima for the gradient-based heatmap $\mathcal{H}_\nabla$. (Color figure online)

2.3 Modification and Post-processing

Similar to [16], an approach based on image pyramids has been implemented. In order to keep the results section concise, we restrict ourselves to only one downsampling step p, 2 to 8. The searching range is, logically, diminished by the factor $1/p$ which allows to perform a coarse registration in around $1/p^2$ of time. The subsequent fine registration uses the original resolution and the searching step has now the order of magnitude of the pyramid size. Note that doing so, a failed coarse registration cannot be corrected during fine registration. To cope with this, we firstly wish to avoid getting stuck in a local minimum by performing exhaustive search as described above; moreover, we choose quite neutral parameter values, such as $\alpha = 0.5$ in (1) and $w_\nabla = 0$ outside in (2). The fine registration takes place using Nelder-Mead method with varying parameters, whereby the starting value at original resolution is computed via nearest neighbor interpolation; that is, in our case $p \arg\min_{\phi'}(\mathcal{H}(\phi'))$. Secondly, we hope that a gross misalignment will only happen to a few isolated buildings such that the upcoming post-processing step will correct these outliers.

For post-processing, centers of gravity of buildings are computed and n nearest neighbors are identified for each of them. This is a fast, almost linear step even for a large number of buildings. Now, median values for offsets in x and y direction are taken from the set of neighbors for every building in order to update the current value. Of course, this step will only improve the performance if the reason for misalignment is justified by our model assumption that close-by buildings have similar offsets. This happens, for example, if a slightly non-nadir view has been taken from a scene containing buildings of approximately similar height. Clearly, in absence of the heatmap proposed e.g. by [14], this strategy of correcting gross errors may have a negative side effect of occasional smearing the discrepancies in the offsets. However, it must be pointed out that both heatmap computation and suitable optimization framework, e.g. by [6], especially with a large number of labels for the random variable, are more costly than the Nelder-Mead optimization and the proposed post-processing step. The value of n chosen for our experiments is 4.

3 Results

This section is structured as follows: In the first paragraph, we will present the datasets and evaluation metrics. Next, quantitative evaluation is provided. Algorithm parameters are varied allowing graphical visualization and interpretation of the results. Also, comparison with previous approaches [14] and [10] is carried out. The last two paragraphs of this section are dedicated to qualitative results and remarks on computation time.

In order to demonstrate the universality of our pipeline, we wish to consider several datasets having completely different properties. Firstly, the settlement called Marco Island, Florida, USA, represents a post-event high resolution data, captured from the air after the Irma hurricane in 2017. The thus captured images were mosaicked without consideration of the non-nadir perspective. Because of the high resolution (5 cm), the roof structures appear quite inhomogeneous; moreover, there are several damaged or destroyed buildings. The offsets are quite high, up to 60 pixels or 3 m. In total, there were 13 tiles from which we considered two and used 103 and 156 buildings, respectively, which are alignable by a pure translation. We refer to these datasets as D1 and D2. The third dataset, recorded at a much coarser resolution 0.5 m, is a densely built region in Perth (Australia), called City of Melville. The shapefile, which contains some 1500 buildings, is quite obsolete; hence, 154 buildings were selected for evaluation. The offsets are between -5 and 8 pixels whereby a moderate bias was kept intentionally. This dataset is denoted as D3. In all datasets, the offsets were measured manually. To measure the accuracy of our algorithm, we compared the ground truth offsets with the manually measured ones recording the widely used root mean square (rms) error. That is, L_2 norms of deviations were computed, averaged and shown in Fig. 2 as variable δ for each set of parameters. Additionally, in datasets D1 and D2, we differentiated between damaged and non-damaged buildings. As for the parameters, we varied α between 0 (only gradient-based) and 1 (only color-based) as well as w_∇ and $w_\mathbf{f}$ in dataset D3 (since infrared channel was available) according to the choices from (2).

The first observation one can immediately derive from the graphs in Fig. 2, top, is that for good parameter sets, the errors in relative deviations can be reduced below 6 pixels in dataset D1 and even below 4 pixels in D2, corresponding to 0.3 or 0.2 m, respectively. What is not recorded in the graphs is that the *distribution* of errors (medians of deviations below 1 pixel) indicates that there are some outliers degrading the performance. Unfortunately, these outliers are often those buildings with damaged roofs as we indicated by circles: the larger the radius, the larger the ratio between average inaccuracies over all damaged and all non-damaged buildings. The most dramatic ratios of almost 2.0 tend to be obtained for the choices of $w_\nabla(\mathbf{p}) \neq 0$ outside of the building mask without pyramids (red curves) while for green curves, the changes between accuracy over damaged or non-damaged buildings are more or less statistical (0.7 to 1.1 with pyramids for D2). The reason is that occasional gradient discontinuities within the roofs of destroyed buildings lead to confusion with usual texture elements outside. Even though the best results were obtained without pyramids, the strategy of applying the Nelder-Mead method on high resolution data seems to be a risky business because the balance parameter α must be chosen with care. For the strategy based on pyramids, the course of the graphs is much more flat. This means that the job of coarse registration has mostly been done well. Notably, the dashed green curve lies below the red one for D1 and above for D2.

Turning our attention to dataset D3 and Fig. 2, bottom, we can see that the rms errors can be reduced to values between 1.5 and 1 pixels and that there are basically two ways to achieve it. First, we could proceed similarly to D1 and D2 by choosing a reasonable α in (1). Alternatively, we could consider the classification result by $C(m)$ in (2) and here it is recommendable to *omit* the gradient-based term. Similarly to D1 and D2, the red and orange curves show

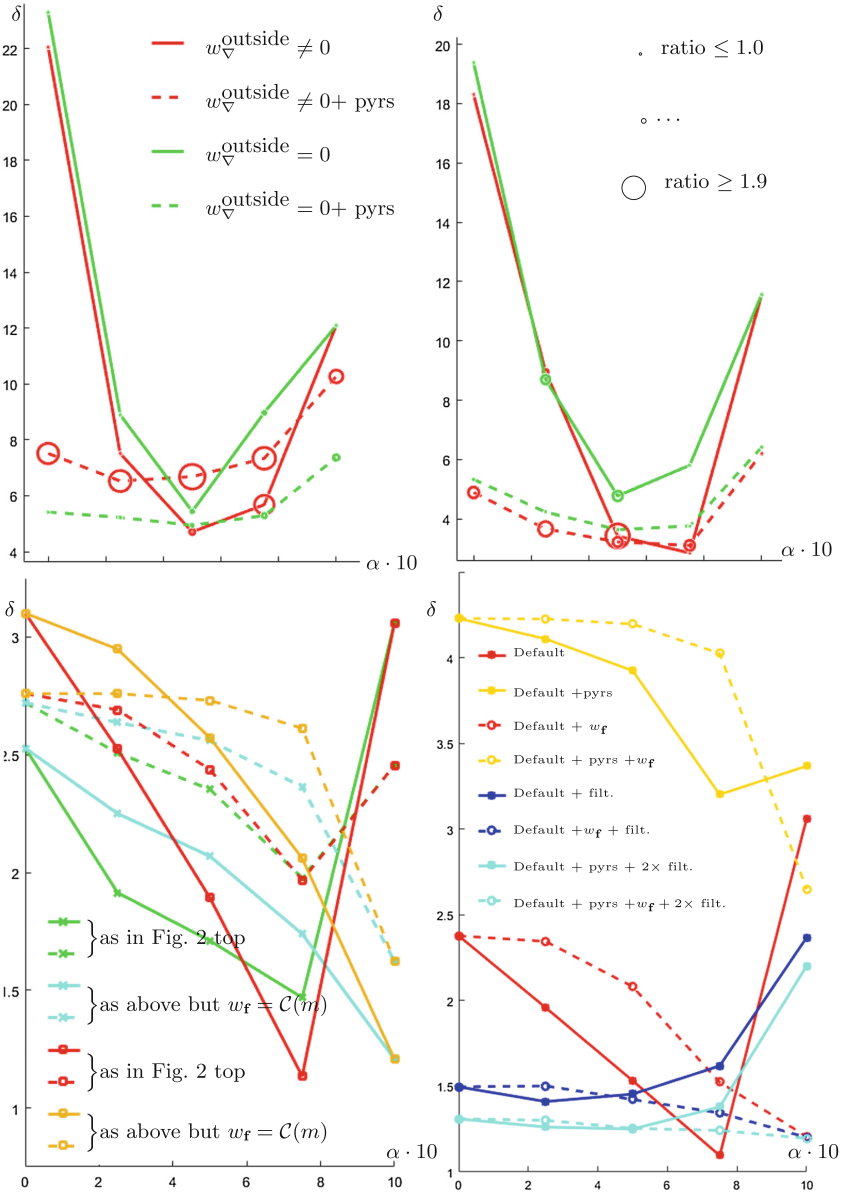

Fig. 2. Top row: results for datasets D1 (left) and D2 (right) depending on the parameter choice. For convenience, α in (1) was multiplied by 10. By circles we indicate to what extent the performance on damaged roofs was worse than for those with non-damaged roofs. Bottom row: results for dataset D3 depending on the parameter choice. Please note that red and orange solid curves in the left image coarsely correspond to red curves on the right while red and orange dashed curves in the left image coarsely correspond to orange curves on the right. See legend for more details. (Color figure online)

a sharper minimum (or a higher sensitivity to the balance parameter α) than the green and cyan ones. Differently to the datasets taken at a finer resolution, all configurations perform *worse* if pyramids are applied which gives us a hint that a resolution of 1 meter and coarser could be critical for our method. With respect to the post-processing, we took the red and orange curves in Fig. 2, left, with a little different value of σ, performed registration for *all* $\approx$1500 buildings, and, after applying filtering, recorded the results only for the selected ones. Two reasons to consider the red and orange curves are that they are most promising and, at the same time, they exhibit a sharper minimum with respect to α, which we hope to smooth during post-processing (blue curves). Moreover, if we use pyramids and post-process twice (once after the coarse and once after fine registration), we obtain cyan curves. We see that the range of acceptable values of parameters becomes more stable and that for almost all configurations, blue and cyan curves are lying below their respective red and orange counterparts. The last observation, not recorded in Fig. 2, concerns application of (1) without term $\|\mathcal{I}_\mathbf{f}\|$. The results become worse especially for quite large and small α, which means that outlines occasionally jumps to a neighboring building. In other words, the classification result alone could be insufficient.

To compare our approach with other state-of-the-art procedures, e.g. [14] and [10], we should first mention that they worked with different (to each other and to us) and geographically hardly comparable datasets, had different problem settings and evaluation metrics. The precision and recall of [14] were 80% and 64% respectively. However, since our final offsets are given in pixels, we must take into account the average building size and make some simplification assumptions. Assuming that the area of a roof is $12 \times 12 = 144\,\mathrm{m}^2$ and that the offsets in both x and y are $0.5/\sqrt{2} \approx 0.35\,\mathrm{m}$ (to yield a not unrealistic result of 0.5 m rms), both precision and recall in our approach would be approximately $(1 - 2 \cdot 0.35/12)^2 \approx 0.88$ in the worst case. This is slightly better than their values and at each case can be considered as a good result since some buildings are destroyed and the running time is less than half. The intersection over union, used by [10], ranges between 0.65 and 0.84 while ours would be 0.79 in this worst case.

For qualitative results, we refer to Fig. 3, where the overall good performance and, at the same time, some few shortcomings of the proposed method can be observed. In Fig. 3, top left, we see why post-processing is not recommendable for D1 and D2: Sometimes, the composition of tiles in the orthophoto is not accurate, but there seem to be other error sources. For example, two neighboring utmost-left buildings are similar in their roof shape, but the offset between the input shape (red) and the manually clicked ground truth (green) differ dramatically. Out of nine building in this fragment, seven were registered successfully. In the top right image, the building marked by 1 exemplifies how the gradient-based term in (1) was deceived by the footpath which has a similar color to the roof. The building was aligned to the border between this path and grass area. Moreover, the image shows a successful registration of a widely non-damaged (2) and severely damaged building roof (3). For dataset D3, in Fig. 3, bottom, we see how buildings 1 and 2 could be correctly aligned after taking into account

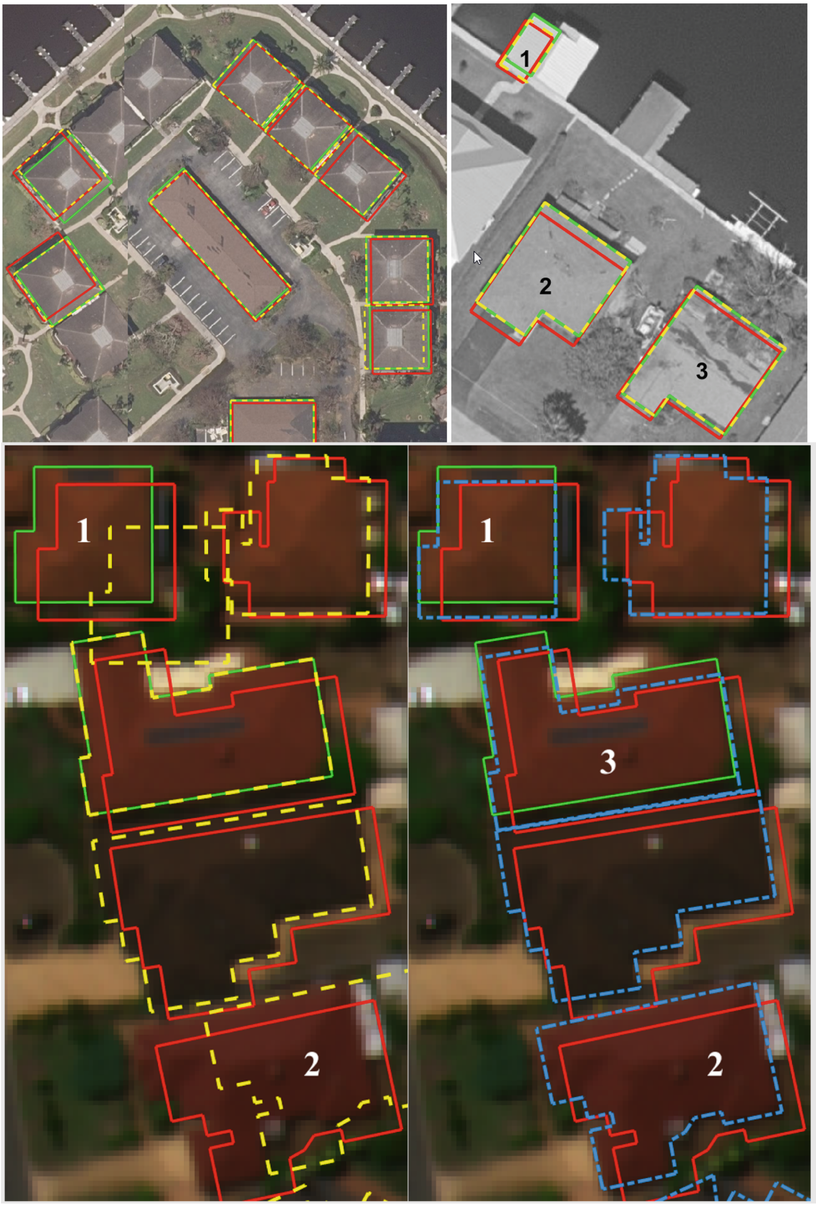

Fig. 3. Top row: example fragments of datasets D1 (on the left) and D2 (on the right, as gray image). Manually measured ground truth, input outlines, and results of our algorithm are specified by green, red, and dashed yellow lines, respectively. Bottom row: example fragment of dataset D3. Here, results of our algorithm before and after post-processing are specified by dashed yellow lines on the left, respectively, dashed blue lines on the right. For more details (numbers), see text. (Color figure online)

the offsets of the surrounding buildings in the post-processing step. At the same time, the result for building 3 on the right became slightly degraded.

From the point of view of computing time, for datasets D1 and D2, averagely 5.25 and 9 s are needed per building, whereby the approach was run on a standard PC with default parameters and without pyramids. If pyramids are used, the computation times for coarse and fine registration are 2.75 s resp. 4.25 s for D1 and 3.2 s and 7.7 s for D2. Clearly, the lions' share for the computation time is made up by the repeated evaluation of our cost function for energy minimization. This explains why the optimization using pyramids does *not* run in a lower time: this number of evaluations does not grow with the resolution since the method is not pixelwise. Only because of better initial values, less internal iterations are needed. As for the exhaustive approach, we noted that while the convolutional operators work efficiently until a certain matrix size (2.75 s per building for the pyramid step $p = 8$), the computation cost already explodes for the next level, $p = 4$, to 170 s per building. The computing type for D3 measures less than 1.5 s per building and the time spent on the post-processing step was negligible. It remains to say that the current MATLAB code was only optimized algorithmically, but not computationally.

4 Conclusions

We presented an approach for adjustment of building outlines stemming from the GIS data with quasi-nadir aerial images. This approach was developed and successfully tested for a real-case application, namely, roof damage assessment after a severe disaster. Its core procedure is based on minimization of an energy function consisting of two terms linked by balance parameter. In all datasets, reasonable values of α allow to achieve best results, which were around 0.2 m and almost 0.5 m for datasets having a finer and coarser resolution, respectively. Another valid conclusion is that depending on how to penalize the gradient outside the building area, more or less stable the curves are with changing balance parameter α. One may think that the algorithm has quite many parameters: b (number of color histogram bins), α in (1), choices of $w_{\mathbf{f}}$, and w_{∇}, σ in (2), number of searching locations for energy minimization and that of surrounding buildings (n) in Sect. 2.3. However, our modules for pre-processing (coarse registration at a lower scale) and post-processing (outlier suppression using a median-filter-based approach) allowed to keep the results stable for a wide ranges of these parameters, as dashed curves in Fig. 2, top and bottom left, as well as blue and cyan curves in Fig. 2, bottom right, show.

We could see that even a quite basic classification result based on NDVI not only strongly improves the results but also the makes gradient-based term widely unnecessary. This is an important conclusion: nowadays context-free CNN approaches based on nested gradient computation are very popular. However, just a little of context helps to obtain good results without training data at all. Therefore, more effort must be put in the future to improve the classification. Here, CNNs will show themselves more than helpful.

For the dataset at a coarse resolution, the pyramid-based approach has turned out to be less successful. It would be, however, interesting to explore how the heatmap can be exploited: either upsampled to a finer resolution using a higher degree polynomial, instead of the currently used Nearest Neighbor Interpolation, or considered it as a data term in non-local energy minimization framework.

Acknowledgment. We thank to City of Melville and, in particular, Dr. Petra Helmholz from Curtin University, for providing the dataset D3.

References

1. Benedek, C., Descombes, X., Zerubia, J.: Building detection in a single remotely sensed image with a point process of rectangles. In: Proceedings of International Conference on Pattern Recognition (ICPR), pp. 1417–1420. IEEE (2010)
2. Brooks, R., Nelson, T., Amolins, K., Hall, G.B.: Semi-automated building footprint extraction from orthophotos. Geomatica **69**(2), 231–244 (2015)
3. Champion, N., Stamon, G., Deseilligny, M.P.: Automatic GIS updating from high resolution satellite images. In: International Conference on Machine Vision Applications (MVA), pp. 374–377. Citeseer (2009)
4. Haklay, M.: How good is volunteered geographical information? A comparative study of OpenStreetMap and Ordnance Survey datasets. Environ. Plann. B Plann. Des. **37**(4), 682–703 (2010)
5. Hermosillo, G., Chefd'Hotel, C., Faugeras, O.: Variational methods for multimodal image matching. Int. J. Comput. Vis. **50**(3), 329–343 (2002)
6. Hirschmüller, H.: Stereo processing by semi-global matching and mutual information. Trans. Pattern Anal. Mach. Intell. **30**(2), 328–341 (2008)
7. Lagarias, J.C., Reeds, J.A., Wright, M.H., Wright, P.E.: Convergence properties of the Nelder-Mead simplex method in low dimensions. SIAM J. Optim. **9**(1), 112–147 (1998)
8. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
9. Lucks, L., Pohl, M., Bulatov, D., Thönessen, U.: Superpixel-wise assessment of building damage from aerial images. In: International Conference on Computer Vision Theory and Applications (VISAPP), pp. 211–220 (2019)
10. Marcos, D., et al.: Learning deep structured active contours end-to-end. In: Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8877–8885. IEEE (2018)
11. Peng, J., Zhang, D., Liu, Y.: An improved snake model for building detection from urban aerial images. Pattern Recogn. Lett. **26**(5), 587–595 (2005)
12. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
13. Tasar, O., Maggiori, E., Alliez, P., Tarabalka, Y.: Polygonization of binary classification maps using mesh approximation with right angle regularity. In: International Geoscience and Remote Sensing Symposium (IGARSS). IEEE (2018)
14. Vargas-Muñoz, J., Marcos, D., Lobry, S., Dos Santos, J.A., Falcão, A.X., Tuia, D.: Correcting misaligned rural building annotations in open street map using convolutional neural networks evidence. In: International Geoscience and Remote Sensing Symposium (IGARSS), pp. 1284–1287. IEEE (2018)

15. Vetrivel, A., Gerke, M., Kerle, N., Nex, F., Vosselman, G.: Disaster damage detection through synergistic use of deep learning and 3D point cloud features derived from very high resolution oblique aerial images, and multiple-kernel-learning. ISPRS J. Photogramm. Remote Sens. **140**, 45–59 (2018)
16. Zampieri, A., Charpiat, G., Girard, N., Tarabalka, Y.: Multimodal image alignment through a multiscale chain of neural networks with application to remote sensing. In: European Conference on Computer Vision (ECCV) (2018)