

# Automatic Feature-Based Geometric Fusion of Multiview TomoSAR Point Clouds in Urban Area

Yuan Yuan Wang, *Student Member, IEEE*, and Xiao Xiang Zhu, *Senior Member, IEEE*

**Abstract**—Interferometric synthetic aperture radar (InSAR) techniques, such as persistent scatterer interferometry (PSI) or SAR tomography (TomoSAR), deliver three-dimensional (3-D) point clouds of the scatterers' positions together with their motion information relative to a reference point. Due to the SAR side-looking geometry, minimum of two point clouds from cross-heading orbits, i.e., ascending and descending, are required to achieve a complete monitoring over an urban area. However, these two point clouds are usually not coregistered due to their different reference points with unknown 3-D positions. In general, no exact identical points from the same physical object can be found in such two point clouds. This article describes a robust algorithm for fusing such two point clouds of urban areas. The contribution of this paper is finding the theoretically exact point correspondence, which is the end positions of façades, where the two point clouds close. We explicitly define this algorithm as “L-shape detection and matching,” in this paper, because the façades commonly appear as L-shapes in InSAR point cloud. This algorithm introduces a few important features for a reliable result, including point density estimation using adaptive directional window for better façade points detection and L-shape extraction using weighed Hough transform. The algorithm is fully automatic. Its accuracy is evaluated using simulated data. Furthermore, the proposed method is applied on two TomoSAR point clouds over Berlin with ascending and descending geometry. The result is compared with the first PSI point cloud fusion method (S. Gernhardt and R. Bamler, “Deformation monitoring of single buildings using meter-resolution SAR data in PSI,” *ISPRS J. Photogramm. Remote Sens.*, vol. 73, pp. 68–79, 2012.) for urban area. Submeter consistency is achieved.

**Index Terms**—Point cloud fusion, SAR tomography (TomoSAR), synthetic aperture radar (SAR), TerraSAR-X.

Manuscript received May 30, 2014; revised August 22, 2014; accepted September 18, 2014. This work was supported in part by the International Graduate School of Science and Engineering, Technische Universität München (Project 6.08: “4D City”), in part by the Helmholtz Association under the framework of the Young Investigators Group “SiPEO” (VH-NG-1018, [www.sipeo.bgu.tum.de](http://www.sipeo.bgu.tum.de)), in part by the German Research Foundation (DFG, Förderkennzeichen BA2033/3-1), and in part by the Gauss Centre for Supercomputing e.V. ([www.gauss-centre.eu](http://www.gauss-centre.eu)) by providing computing time on the GCS Supercomputer SuperMUC at Leibniz Supercomputing Centre (LRZ, [www.lrz.de](http://www.lrz.de)).

Y. Wang is with the Helmholtz Young Investigators Group “SiPEO”, Technische Universität München, Munich 80333, Germany (e-mail: [wang@bv.tum.de](mailto:wang@bv.tum.de)).

X. Zhu is with the Helmholtz Young Investigators Group “SiPEO”, Technische Universität München, Munich 80333, Germany, and also with the Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Weßling 82234, Germany (e-mail: [xiao.zhu@dlr.de](mailto:xiao.zhu@dlr.de)).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JSTARS.2014.2361430

## I. INTRODUCTION

### A. TomoSAR

**T**OMOGRAPHIC SAR inversion (TomoSAR) [2]–[6] is an interferometric synthetic aperture radar (InSAR) technique that tracks down the long-term or partially coherent pixels in a stack of coregistered SAR images, which is similar to other InSAR techniques such as persistent scatterer interferometry (PSI) [7]–[10], small baseline subset (SBAS) [11]–[13], SqueeSAR [14], [15], and CEASAR [16]. They all aim at retrieving the three-dimensional (3-D) position and the parameters of the undergoing motion of point and/or volumetric scatterers.

However, fundamentally different, TomoSAR exploits possible multiple scatterers in the third dimension *elevation* of a rang-azimuth pixel—a phenomenon called *layover* in radar jargon that happens when two or more scatterers are at the same range (distance) to the sensor. For example, the closely packed vertical structures in urban areas cause severe layover. TomoSAR retrieves the elevations and motion parameters of multiple scatterers by means of spectral estimation. This is not possible with the above-mentioned single scatterer model-based methods, e.g., PSI. Therefore, TomoSAR is so far the most competent InSAR method for long-term urban monitoring [17].

Therefore, the 3-D TomoSAR point clouds derived from meter-resolution SAR image stacks are the basis of the work of this paper. The point clouds we used are obtained by *Tomo-GENESIS* [18]—a TomoSAR software of the German Aerospace Center (DLR) for large urban areas monitoring. It is developed based on the work of [5], [6], [17], [19]–[21]. It delivers, for the first time, a 5-D point cloud—3-D position plus linear deformation rate and amplitude of seasonal motion—from a stack of SAR images. The point density is up to  $10^6/\text{km}^2$ , when using TerraSAR-X high-resolution spotlight data. Such density is comparable with some light detection and ranging (LiDAR) product and enables the retrieval of the features of individual building.

### B. Fusion Geometry of Cross-Heading InSAR Point Clouds

Cross-heading orbits refer to satellite's ascending and descending flying trajectories. Due to the side-looking geometry of SAR, minimum of two TomoSAR point clouds from cross-heading orbits are required to achieve a complete monitoring over the whole area. As demonstrated in Fig. 1, point cloud derived from SAR image stacks of ascending or descending orbits provides either the front or the rear side of the same building. Such two point clouds have basically no *exact point*

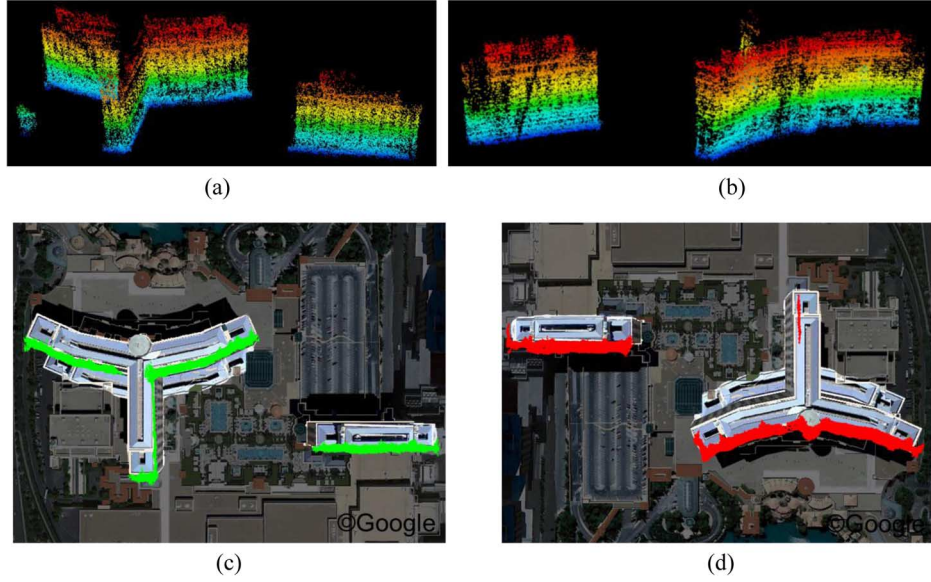


Fig. 1. TomoSAR point cloud derived from (a) an ascending image stack in Las Vegas and (b) a descending image stack of the same area. Color represents the height. (c) and (d) are the corresponding footprint of (a) and (b) on Google Map satellite images.

*correspondence*, which is referred to as the identical geolocation of the sources of two corresponding points (in the two point clouds, respectively). Examples are man-made objects like cylindrical reflector or 3-D corner reflector that could reflect radar signal in all directions. Not difficult to imagine, fusion of such point clouds is as tricky as coregistering two optical images of front and rear side of an opaque object.

These two point clouds are always relative to two different reference points with unknown heights. The uncompensated height of the reference point causes the point cloud to shift along the elevation direction. Therefore, the fusion task is to find the unknown heights of the two reference points. A schematic drawing of fusion in Universal Transverse Mercator (UTM) coordinate is illustrated in Fig. 2 (modified from [1]), where the left and right satellites indicate the ascending and descending master orbits at a fixed *east-up* plane, respectively. Perpendicular toward the reader's screen/paper is the *north* direction. The flying direction of the two satellites is indicated by the cross (fly away from reader) and dot (coming toward reader) on the satellites. They are not parallel to the north direction. The angle between the flying direction and the meridian (north) is the so-called heading angle.  $R^a$  and  $R^b$  are the range distances of the two orbits. The two black dots are the corresponding scatterers assumed to have identical position, but geocoded into different positions  $\mathbf{P}_{xyz}^a$  and  $\mathbf{P}_{xyz}^b$  due to their different and unknown reference height  $\Delta z^a$  and  $\Delta z^b$ . This also causes horizontal shifts  $\Delta xy^a$  and  $\Delta xy^b$ , which are functions of the incidence angles  $\theta^a$  and  $\theta^b$ , i.e., the horizontal shift is not constant throughout the whole point cloud. To achieve a fusion, the two scatterers have to be shifted along their elevation directions by  $\Delta S^a = \Delta z^a / \sin \theta^a$  and  $\Delta S^b = \Delta z^b / \sin \theta^b$ , respectively. The fusion model in UTM coordinate system is then [1]

$$\mathbf{P}_{xyz}^a + \frac{\Delta z^a}{\sin \theta^a} \mathbf{s}^a = \mathbf{P}_{xyz}^b + \frac{\Delta z^b}{\sin \theta^b} \mathbf{s}^b \quad (1)$$

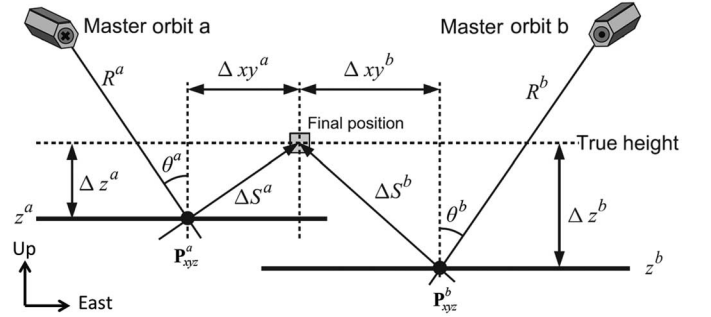


Fig. 2. Schematic representation (modified figure from [1]) of the InSAR point cloud shift in UTM coordinate due to different reference points. The reader's screen/paper is the east-up plane, and perpendicular to it away from the reader is the north direction. The two satellites indicate the ascending and descending master orbits positions. Their flying directions are indicated by the cross (fly away from reader) and dot (coming toward reader) on the satellites. They are not parallel to the north direction. The angle between the flying direction and the meridian (north) is the so-called heading angle. The two black dots are the two scatterers assumed to be identical in UTM coordinate system, but shifted in the two point clouds due to the heights  $\Delta z^a$  and  $\Delta z^b$  of their reference points.

where bold letters indicate vectors, and  $\mathbf{s}^a$  and  $\mathbf{s}^b$  are the unit vectors of the elevation directions of the two scatterers, which depends on the incidence angle  $\theta$ , and heading angle  $t$

$$\mathbf{s} = \begin{bmatrix} \cos t \cos \theta \\ -\sin t \cos \theta \\ \sin \theta \end{bmatrix}. \quad (2)$$

In another words, three equations can be written for each pair of corresponding points ( $\mathbf{P}_{xyz}^a, \mathbf{P}_{xyz}^b$ )

$$\begin{aligned} x^a - x^b + \Delta z^a \cos t^a \cot \theta^a - \Delta z^b \cos t^b \cot \theta^b &= 0 \\ y^a - y^b - \Delta z^a \sin t^a \cot \theta^a + \Delta z^b \sin t^b \cot \theta^b &= 0 \\ z^a - z^b + \Delta z^a - \Delta z^b &= 0 \end{aligned} \quad (3)$$



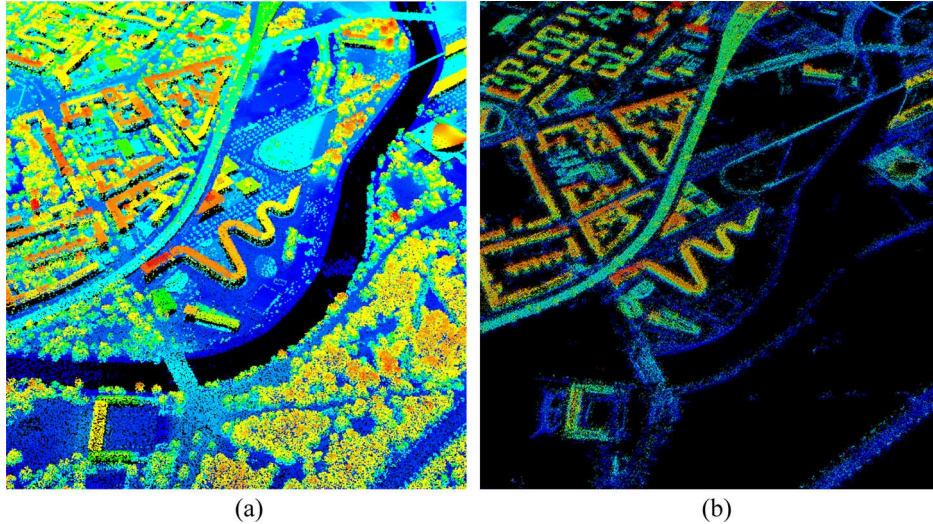


Fig. 3. Comparison of (a) LiDAR point cloud [22] and (b) TomoSAR point cloud of the same area in downtown Berlin. The color represents the height. LiDAR point cloud has, in general, better quality in terms of 3-D positioning accuracy.

where  $\Delta z^a$  and  $\Delta z^b$  are the only unknowns. However, the incidence and heading angles  $\theta$  and  $t$  are direct functions of the point current position. As described in [1], this problem is the so-called *Gauß-Helmert* model. It needs to be solved iteratively, where  $\mathbf{P}_{xyz}^{\{a,b\}}$ ,  $\theta^{\{a,b\}}$ , and  $t^{\{a,b\}}$  are updated at each iteration. The estimation is robust with many pairs of points available.

### C. Related Work and Motivation

In computer vision, the 3-D rigid point cloud coregistration methods are very well studied. For a successful coregistration, overlapping of the two point clouds is required, so that one can find common points or features that describe a common location in the point clouds. The basic approach of such task follows the procedures of feature detection, description, and matching, where so do the researches focus. The feature detection and description are usually combined. The most well-known ones are:

- 1) gradient-based features, e.g., scale invariant feature transform (SIFT) [23] and speeded up robust features (SURF) [24];
- 2) histogram-based features, e.g., histogram of gradients (HoG) [25] and point feature histogram (PFH) [26]; and
- 3) surface normal-based features, e.g., spin-image [27] and normal-aligned radial feature (NARF) [28].

Here, we just name a few. Many of their variants are also highly used. The matching algorithms, according to the optimization methods used, can be classified as either global or local. As pointed out in [26], the most famous algorithms of the first type are generic and evolutionary algorithms [29], [30]. Not surprisingly, most of the study is the second type, due to the computational complexity of the first type. The most well-known one is the iterative closest point (ICP) [31], [32]. Other highly cited works are mostly variants or improvements in the original ICP. For instance, to find better point correspondence using feature descriptors [26], [33], [34], to find better initial

transformation [34], [35], and to use nonlinear optimization methods in ICP [36], [37].

In the remote sensing community, these techniques are employed for coregistration of LiDAR or optical image derived point clouds [38]–[42]. However, no attempt has been made for fusion of PSI or TomoSAR-derived point clouds of urban areas. This is especially true for two point clouds generated from data stacks of cross-heading orbits, due to the lack of exact point correspondence. Another reason prevents the above-mentioned methods from being directly applicable is the relatively worse accuracy of InSAR point clouds compared with LiDAR or optical image-derived ones. An example is made in Fig. 3, which is the comparison of LiDAR and TomoSAR point cloud of the same area in Berlin.

The first attempt of such task is described in [1] using PSI point clouds. It is essentially a 3-D surfaces matching that relays on removing the façade points, where the point correspondence is most unlikely to appear. The point correspondences are found by searching closely space point pairs on surface. Its accuracy is limited, since the exact point correspondence is not addressed. The computational efficiency is compromised when applying to TomoSAR point clouds [43]. Practically, only 10% of the non-façade points are used, since a TomoSAR point cloud is usually much denser than PSIs’.

Therefore, the initiative of the proposed method is to make use of the rich façade information in TomoSAR point clouds in order to derive the theoretically exact point correspondence and to develop a computationally efficient algorithm that can handle large point clouds. Finally, we can obtain a 3-D point cloud associated with the movement information covering an entire city.

As no exact point correspondence can be found among the points of the two point clouds, we will have to look for higher level features. Studies such as [44]–[46] have already suggested that quadrilateral buildings appear as “L-shapes” in SAR images. As a result, the end points of the L-shape, i.e., the building edges where the two point clouds close, can be used as the

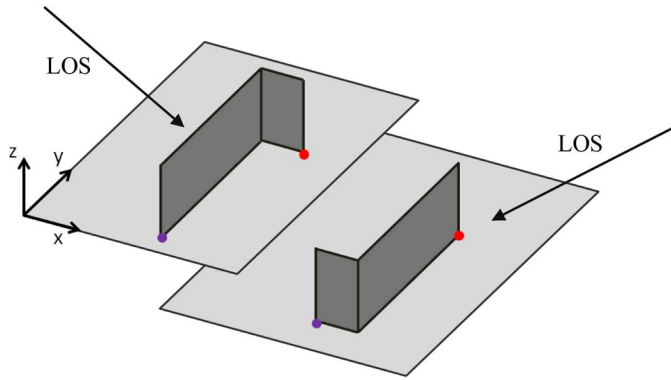


Fig. 4. Fusion geometry of the proposed L-shape method. The two building façades are colored in dark gray representing the geocoded point clouds from cross-heading orbits whose LOS directions are indicated by the two arrows. The light gray planes denote the corresponding ground level of the two point clouds. The two pairs red and purple dots are the façade end point position to be estimated.

exact point correspondence. Described in Fig. 4, the L-shape end points are indicated by the two pairs of red and purple dots. The two building façades colored in dark gray represent the point clouds from ascending and descending orbits whose line of sight (LOS) directions are indicated by the two arrows, respectively. The light gray planes denote the corresponding ground levels of the two point clouds.

The coregistration is then broken down to the task of detecting the L-shapes' end points, estimating their 3-D positions as the descriptor, and finally matching the two reduced point clouds. To achieve this, some robust procedures are designed to extract façade points, estimate the positions of the façade edges, and then find correspondence in the two point clouds.

## II. FEATURE-BASED INSAR POINT CLOUD FUSION

The flowchart of the algorithm is plotted in Fig. 5. Starting from the raw TomoSAR point clouds, the outliers are filtered out using neighborhood analysis. The façade points are then extracted using point density estimation. Subsequently, each point cloud can be split into the façade part and the non-façade part. By cross-correlating their non-façade parts, the two point clouds can be coarsely coregistered within an accuracy of several meters, which gives an initial guess of the final shifting vector. The façade parts of the two point clouds are segmented into building blocks assisted by a two-dimensional (2-D) building shape layer from online map provider such as OpenStreetMap. After obtaining the segmented point clouds, L-shape is searched for each segment, and the positions of the end points of the L-shape are estimated. Therefore, the fusion of two complete point clouds are reduced to the fusion of two much sparser point clouds comprised of the virtual façade end points. Thus, the same random sample consensus (RANSAC) point correspondence matching procedures introduced in [1] can be applied. The final shifting vectors are estimated using (3).

### A. Point Cloud Filtering

As mentioned in Section I-C, the overall geolocalization quality of a raw TomoSAR point cloud is subject to the high

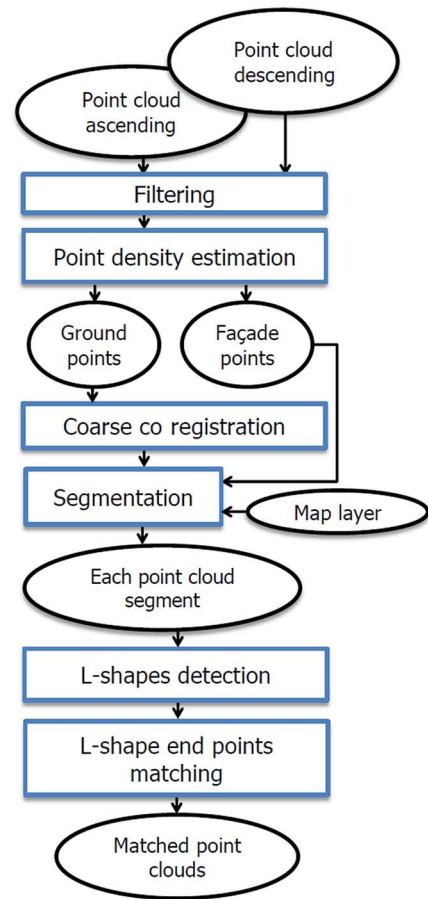


Fig. 5. Flowchart of the proposed algorithm. The black ellipses represent input/output data. The blue rectangles are procedures/processes.

dynamic range of the SAR image pixels' SNR. The criteria in model order selection for rejecting noncoherent pixels are usually set to be very soft [5], in order to preserve as much information as possible from the expensive data stacks. This leads to some amount of outliers in the resulting raw TomoSAR point clouds. Fig. 6(a) is an example of a raw TomoSAR point cloud of Berlin central station. Outliers appear as single point or small clusters elevated from the main cluster nearby. They usually have less neighboring points and experiencing larger distances to their nearest neighbor.

Therefore, the outlier filtering can be done based on the neighborhood analysis of each point in the raw point cloud. We use the mean distance to  $k$ -nearest neighbors as the single criteria, i.e., for each point, we look for  $k$  points that are closest to it, and calculate the mean distance. Points with mean distance larger than a certain threshold are discarded. For instance, in Fig. 7(a), the mean distances of the points in Fig. 6(a) to their 20 nearest neighbors are plotted as a top view. The mean distance is color coded according to the color bar. Closely clustered points have small distances showing in blue, while outliers show larger distances and are usually displayed in red. Fig. 7(b) is the histogram of the  $\log_{10}$  of this mean distance. The histogram shows two major distributions: 1) the left one refers to the main structures, and 2) the right one corresponds to points away from the main structures. The outliers can be removed



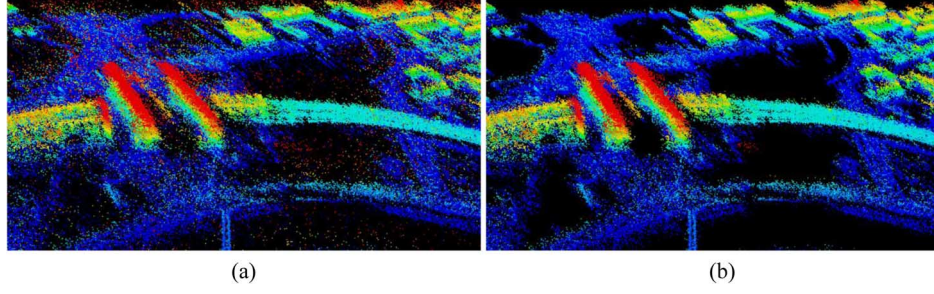


Fig. 6. (a) Raw point cloud of Berlin central station and (b) filtered point cloud. Color represents the height.

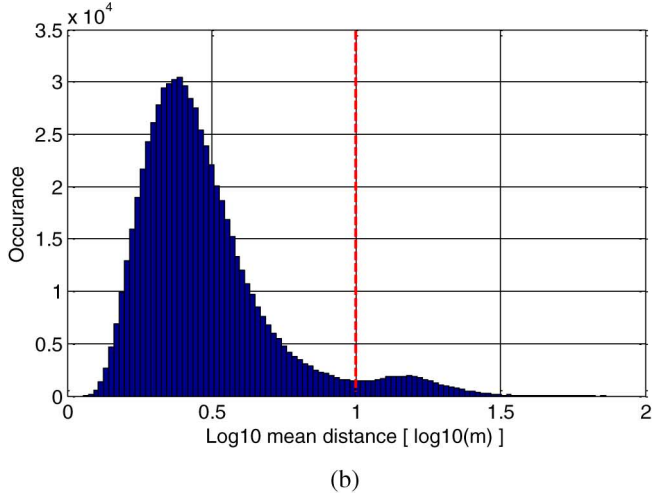
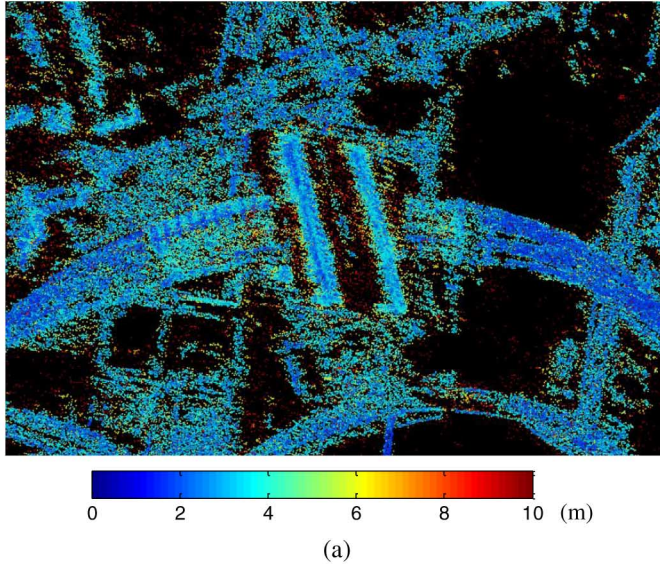


Fig. 7. (a) Mean distances of the points in Fig. 6(a) to their 20 nearest neighbors and (b) the histogram of the log 10 of this mean distance. The red dashed line indicates the cutoff distance which is 10 m.

by thresholding (indicated by the red dashed line) on the mean distance. As shown in the filtered point cloud in Fig. 6(b), most of the outliers are effectively removed. We found a threshold at 10 m and using 20 nearest neighbors are effective for TomoSAR point cloud derived from TerraSAR-X high-resolution spotlight image stacks.

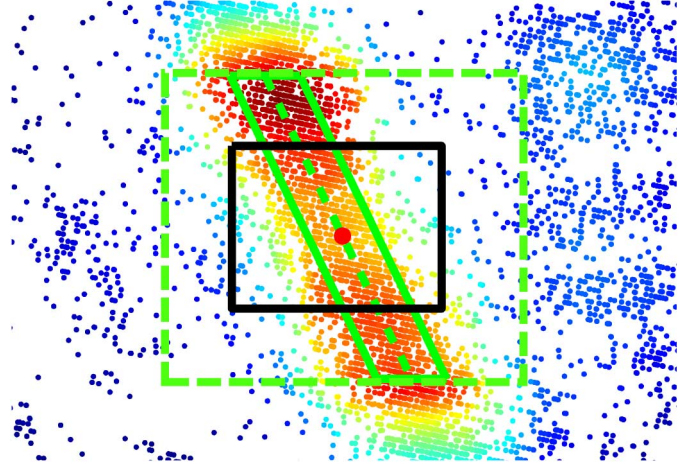


Fig. 8. Schematic drawing of point density estimation using directional window. The colored points are the projection of the 3-D point cloud in *east-north* plane, with the color indicating the point density. Blue indicates low density and red indicates high density. The color bar is not explicitly given for simplicity. The green-dashed line segment is a robust fit to the 2-D projection in a selected area. Points within the region bounded by the solid green lines are considered in the directional point density estimation, while those within the black box are considered in the box-car density estimation.

### B. Point Density Estimation Using Directional Window

The point density per square meter is employed for façade point detection. Because of the vertical accumulation of the points on the façade, their point density is usually much higher over other areas. In order to better distinguish the façade points from other, we introduce a point density estimation method using an adaptive directional window, instead a moving box.

The approach is illustrated in Fig. 8. It shows the projection of a point cloud in the horizontal plane, with the color indicating the point density, and the red dot being the target point whose point density is going to be estimated. The estimation follows three steps: 1) points within an area (the green-dashed rectangle) centered at the target point is selected; 2) a straight line segment (the green-dashed line) passing through the target point is fitted using robust algorithm, e.g., RANSAC [47], robust principle component analysis [48], etc.; and 3) only the inliers (within the solid green parallelogram) of the line segment are counted for the point density, instead of counting all the points in the black rectangle.

For comparison, Fig. 9(a) and (b) shows the point density result using directional and rectangle window, respectively, for

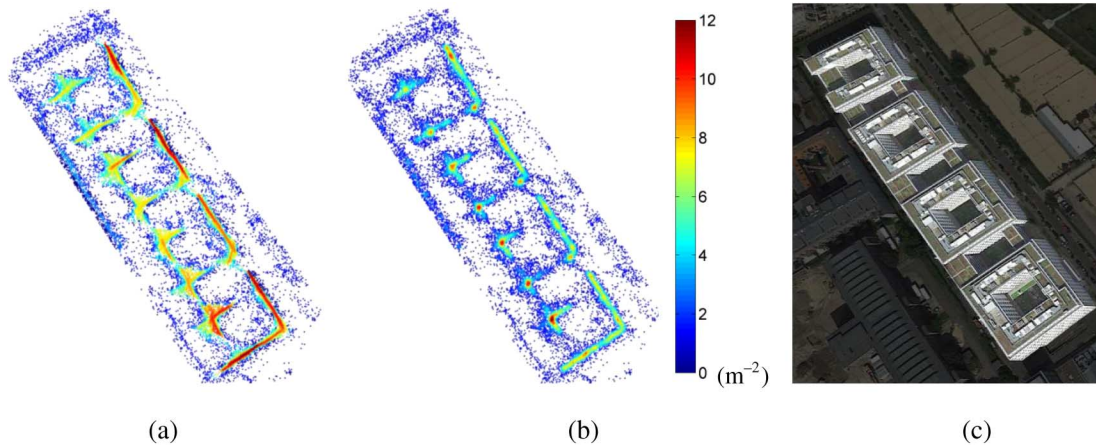


Fig. 9. Comparison of point density estimated using (a) directional window and (b) rectangle window. The color bar applies for both (a) and (b). For reference, (c) is the optical image of the same area from Google. The area of directional and rectangle windows are identical.

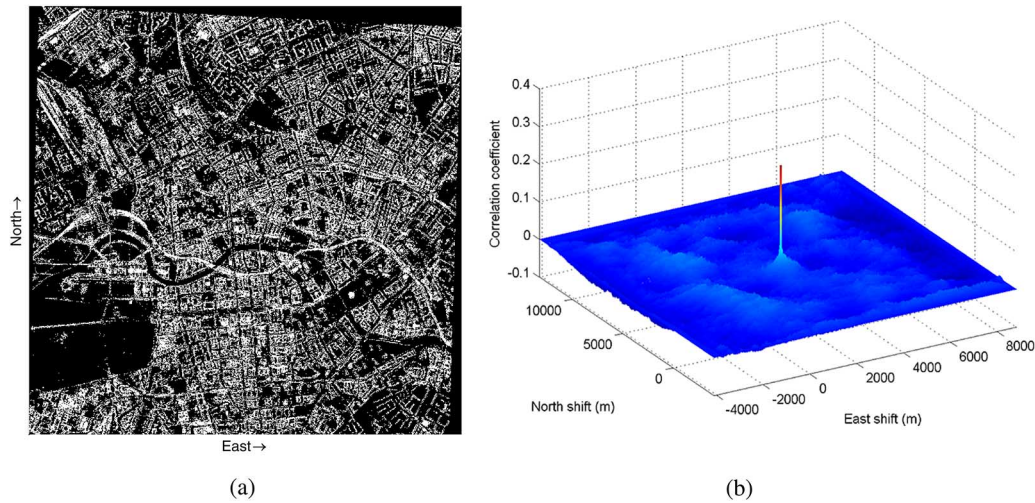


Fig. 10. (a) Binary image of the 2-D footprint of ascending point cloud of Berlin and (b) the 2-D cross correlation coefficient of the footprints of ascending and descending point clouds. The position of the peak value is the estimated coarse shift between the point cloud and the map layer.

a test area near Berlin central station. Since the same color bar is applied to the two figures, it is easy to tell that the directional window emphasizes the façade point, meanwhile keeps the point density of non-façade points unchanged compared with rectangle window. Since the urban buildings mostly consist of linear facades, our method works quite well in emphasizing façade points. The final façade points are selected by thresholding on the point density.

### C. Coarse Coregistration

The point clouds need to be coarsely coregistered to obtain an initial solution. This is a common practice in point cloud coregistration, which limits the search space in the refine coregistration step.

The coarse 3-D coregistration is done separately in horizontal and vertical directions. Horizontally, we cross-correlate the 2-D footprints of the non-façade parts of the two point clouds; and vertically, by simply aligning the mean height of the two point clouds. The 2-D footprints are rasterized binary images with a sampling spacing of a few meters. For example, Fig. 10(a) is

the binary image of the ascending point cloud of Berlin. The descending one will look similar. Their 2-D cross-correlation coefficient is in Fig. 10(b), with brighter pixel being higher correlation. The position of the peak value is the coarse shift amount.

### D. Point Cloud Segmentation

Since L-shape is a local feature of each building, the two point clouds are segmented to building blocks. Since the segmentation only needs to distinguish different building blocks, and it is also not the main focus of the algorithm, we make use of the building shape layer [49] from OpenStreetMap for assistance.

The building shape layer is a binary mask of the buildings [see Fig. 11(a)]. It is naturally not aligned with the point clouds footprints due to the shifted position of the point clouds, and sometimes the incorrect geolocation of the map itself. The alignment of the building shape layer with the point clouds 2-D footprint is done automatically using the same 2-D cross correlating technique explained in the previous section. After the



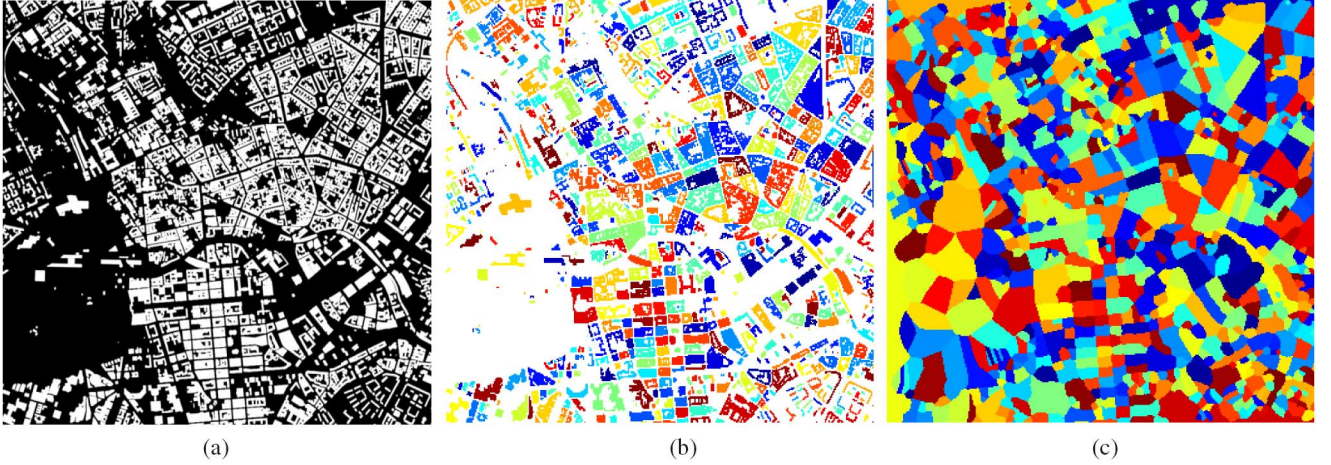


Fig. 11. (a) Building shape layer extracted from OpenStreetMap, (b) labeled individual buildings, and (c) segmentation mask derived from (b). The color assigned to each segment is random.

alignment, each building in the mask is labeled by checking their connectivity, yielding Fig. 11(b), where a random color is assigned to each building for better visualization. The unlabeled areas, i.e., the white region in Fig. 11(b), are assigned to their nearest building, giving the final segmentation mask in Fig. 11(c).

#### E. L-Shape Detection Using Weighted Hough Transform

For each point cloud segment, the façade part is selected, and one or zero L-shape is detected from its projection in the *east-north* plane. The detection is achieved by catching two interconnected line segments using weighted Hough transform. The proposed weighted Hough transform sums up the density of the points inside a Hough bin  $\Delta\theta$  and  $\Delta r$ , instead of only counting the number of points, i.e.,  $H = \sum_{i=1}^N d_i$ , with  $H$  being the Hough transform value,  $N$  the number of points in the Hough bin, and  $d_i$  the point density of each point. This concept is also illustrated in Fig. 12, where the green lines bound the Hough bin, and the size of the points inside the Hough bin denotes their point density. Both  $\Delta\theta$  and  $\Delta r$  are exaggerated for visualization purpose. Standard Hough transform gives the same value for the two bins, but the proposed weighted Hough transform favors the right bin, since its point density is generally higher.

After the weighted Hough transform, the brightest pixel in the Hough matrix is first extracted. Its corresponding line is the most prominent one in the point cloud segment. It is taken as the first line of the L-shape. The neighborhood value of this pixel is suppressed in order to keep a constraint on the minimum angle of the L-shape. Among the rest, a few pixels with the highest amplitude are selected as the candidates of the second line. The second line shall be the one that connects the first one and forms the longest continuous contour. To determine the length of the two line segments, their intersection points are calculated using their equations, and the two end points are selected by locating the last point (with threshold density) passing by the lines, respectively. Nevertheless, such end points position estimation only gives a preliminary estimation of the length of the line segments. Therefore, it needs to be refined for better

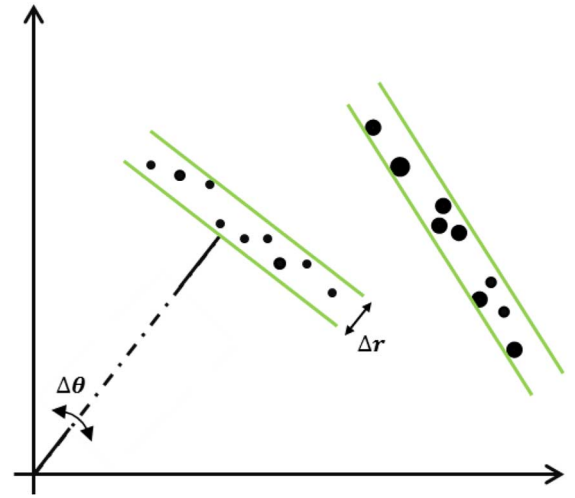


Fig. 12. Illustration of weighted Hough transform. The green arcs bound the region of a Hough bin with spacing of  $\Delta\theta$  and  $\Delta r$ , which is exaggerated for visualization. The dots inside the Hough bins are the point cloud projection in the horizontal plane. The size of the dot represents its point density, with larger size higher the density. Both Hough bins have nine points, which gives the same value in Hough transform. However, the right bin has higher value in weighted Hough transform.

accuracy, which is explained in the Section III. The L-shape detection is applied on each segment of the two point clouds.

Global constraints are also put on

- 1) the minimum line segment length  $l_{\min}$ , according to the knowledge of minimum façade length;
- 2) the minimum Hough value  $H_{\min}$ , which can be calculated as  $H_{\min} = \Delta r l_{\min} d_{\min}^2$ , where  $d_{\min}$  is the minimum point density of an identifiable façade in the point cloud; and
- 3) the opening direction of the L-shape depending on the orbit.

As an example, Fig. 13 shows the detected line segments and the final L-shape of the same point cloud used in Fig. 9. The detected lines are overlaid on the point density. In Fig. 13(a), the longest line segment is fixed as the first line of the L-shape, with the rest being the candidates of the second line segment.

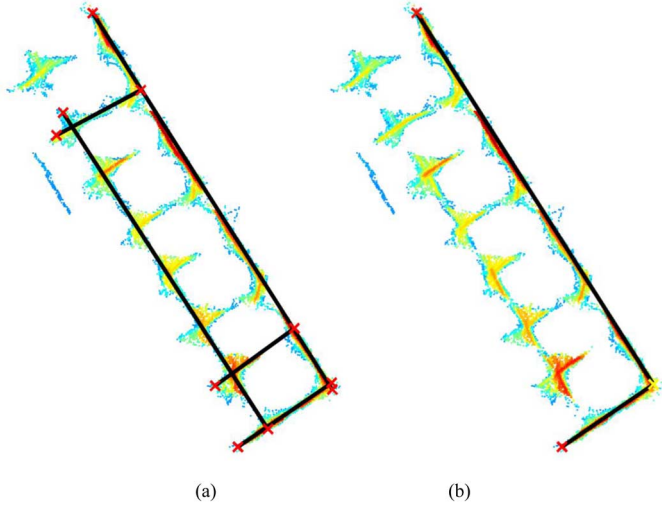


Fig. 13. Example of (a) detected line segments and (b) the final L-shape, using the same point cloud as in Fig. 9. The black lines are detected line segment. The line segments are overlaid on the point density. The roughly estimated end points are marked as red crosses. The estimation of their precise positions is explained in the next section.

Among them, the proposed algorithm automatically determines the second line by checking the connectivity with the first line and the total length of the L-shape. In the final detected L-shape, the intersection point is marked as yellow cross and the end points are marked as green.

#### F. Precise L-Shape End Points Position Estimation

The position of L-shape end point includes its horizontal 2-D coordinates and its height. The height is defined as the position where the façade intersects the ground plane, as depicted already in Fig. 4.

The horizontal coordinates are estimated using a model-based approach. Based on the detected L-shapes, the inlier points of the L-shape are projected to their corresponding line segments, i.e., façade direction. That is to say, we select only the points within a certain perpendicular distance to the L-shape and separate the two arms of the L-shape. One arm of the L-shape with inlier point should look like Fig. 14(a), where the black line is the estimated façade direction from the weighted Hough transform. For each arm of the L-shape, a projection should be obtained. The projection is basically the one-dimensional (1-D) coordinate of the façade points together with the non-façade points in the façade direction. Fig. 14(b) is the projected coordinates of the inlier points in Fig. 14(a). The middle section with much higher density belongs to the façade part. Therefore, its point density can be modeled as a rectangle function plus a constant offset for most of the urban façade.

An estimate of its point density can be obtained by convolving it with a rectangle window. For example, Fig. 14(c) shows the point density estimate of the points shown in Fig. 14(b). Its system model can be written as

$$f(x) = \int \left( A \text{rect} \left( \frac{\tau - a}{2b} \right) + c \right) \text{rect} \left( \frac{x - \tau}{2W} \right) d\tau \quad (4)$$

where  $f(x)$  is the estimated point density as a function of the position  $x$  in the façade direction;  $A$ ,  $a$ ,  $b$ , and  $c$  are the unknown parameter of the exact point density model, where  $A$  models the façade point density,  $a$  is the façade center position,  $b$  is the half façade length, and  $c$  is the point density of the non-façade part. Finally,  $W$  is the known half-width of the rectangle filter. Since the convolution of two rectangle functions is a trapezoid, (4) can be written as

$$f(x) = \begin{cases} 2cW & x \leq a - b - W \\ A(x - a + b + W) + 2cW & a - b - W < x \leq a - b + W \\ 2(A + c)W & a - b + W < x \leq a + b - W \\ -A(x - a - b - W) + 2cW & a + b - W < x \leq a + b + W \\ 2cW & x > a + b + W. \end{cases} \quad (5)$$

Equation (5) shows that the left and right sides of the trapezoid can fully characterize the unknown parameters  $A$ ,  $a$ , and  $b$ . Therefore, a moving fitting method is introduced to robustly fix the equation of the two sides. We use a moving window with the same width as the rectangle filter. A straight line is fitted to the points within the window. For any trapezoidal shape, high slopes should be detected at the position of the left and right sides. By detecting the position of the two high slopes, one can determine the position of the two sides. For robustness, the product of the absolute value of the slope and the number of inliers is considered in the detection of the two sides, because the maximum number of inliers should be reached when the window centered at the left or the right side. Fig. 14(d) shows the product of the absolute value of the slope and the number of inliers for the point density curve in Fig. 14(c). Two prominent peaks can be detected. The positions of the peaks are the direct estimates of the center positions of the left and right sides that are equal to  $a - b$  and  $a + b$ . Hence, they are also the estimates of the start and end position of the rectangle function that models the façade.

To ensure reliable estimation, two regularizations are performed: 1) the coarse façade length obtained from the weighted Hough transform is used as a prior to regulate the estimation; and 2) the difference in the absolute value of the two slopes is restricted, because they must be both close to  $A$  according to the model.

After obtaining the 2-D horizontal coordinates of the L-shape end point, we look for the neighboring points of the 2-D position in the ground level. The mean value of these points is an estimate of the intersection of the 2-D coordinates with the ground plane. It is taken as the height of the L-shape end point. A plane should be fitted to these points, if the local topography is not removed.

#### G. End Points Matching and Final Shift Estimation

At this step, a number of L-shape end points are found for both point clouds. Therefore, the fusion of two complete point clouds are reduced to the fusion of two much sparser 3-D point clouds comprised of only the façade end points. For robustness, RANSAC is employed to find the maximum number of matched point pairs, which is also used in [1]. More details can be found in [50]. Briefly speaking, at each iteration of



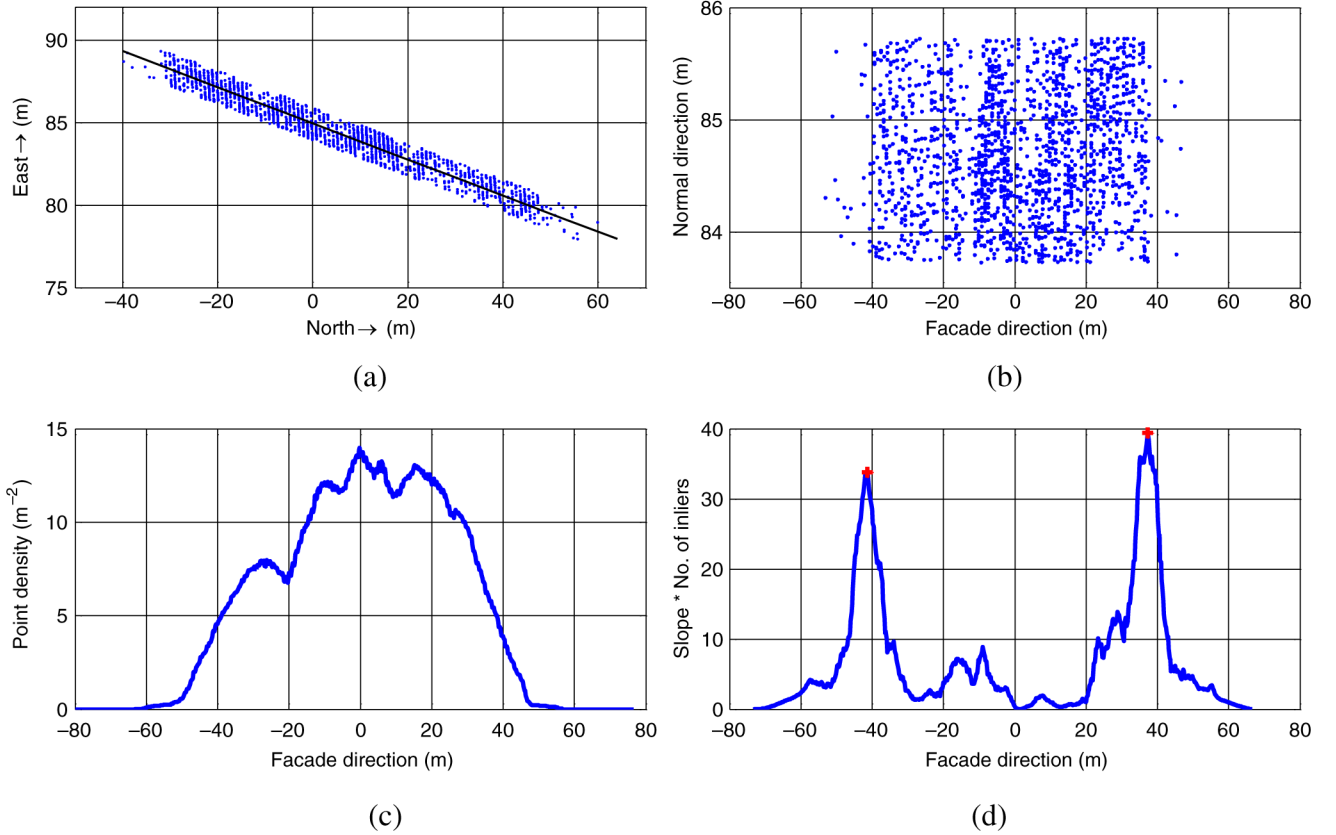


Fig. 14. (a) The inlier points of one arm of an L-shape, with the black line being the façade direction obtained from the weighted Hough transform, (b) the coordinates of the inlier points projected to the façade direction, (c) the estimate of (b)'s point density using a rectangle window, and (d) the product of the slope and the number of inliers of the fitted line using the points in the moving rectangle window along the point density estimate shown in (c). The two red crosses are the automatically detected façade start and end position at  $-41.38$  and  $37.22$  m, respectively.

RANSAC, a pair of façade end points are selected from the two sparse point clouds, least square adjustment following (3) are performed, obtaining the solution of two height offsets  $\Delta z^a$  and  $\Delta z^b$ . The two sparse point clouds are then shifted along their elevation directions. The number of matched pairs, i.e., pairs of points at close range, is then counted. The final shift is given by the one with the maximum number of matched pairs.

### III. DISCUSSION

#### A. Fusion Accuracy

The final fusion accuracy directly depends on the number of matched pairs  $N_{\text{pairs}}$  of façade end points, and the accuracy  $\sigma_{\text{point}}$  of these end point estimates, i.e.,  $\sigma_{\text{final}} = \sqrt{2}\sigma_{\text{point}} / \sqrt{N_{\text{pairs}}}$ , where  $\sqrt{2}\sigma_{\text{point}}$  is the accuracy of one pair of façade end points, according to the fusion model.

$N_{\text{pairs}}$  depends on the quality of the façade end point position estimates which are jointly affected by the accuracy of input TomoSAR point clouds, the segmentation performance, the L-shape extraction accuracy, and the number of quadrilateral buildings in the scene. According to our experience using high-resolution TerraSAR-X data, the number of quadrilateral buildings in the scene plays the most crucial role. Their typical range is a few hundreds for European cities like Berlin. We will exemplify this using real data in Section IV-B.

To analyze accuracy  $\sigma_{\text{point}}$  of the façade end points estimation algorithm explained in Section II-F, we test it using simulated data. It is generated by distributing points along the façade direction, with a constant point density on the façade part, and also a constant, but much lower point density on the non-façade part. The simulated data should look like Fig. 14(b). Important to note that the elevation accuracy  $\sigma_s$  (projected on façade direction) of the TomoSAR point cloud will affect the spatial distribution of the points. In the extreme case, when the TomoSAR point cloud is noise free, the distribution of the points should be evenly spaced in the façade and non-façade part, respectively.

We set total data length to be 40 m, façade length to be 20 m spanning from 12.00 to 32.00 m. The façade points are simulated five times, with the density being 5, 10, 15, 20, and 25/m<sup>2</sup>, respectively. The point density of non-façade area is always kept at 1/m<sup>2</sup>, and the rectangle filter size to be 5 m. Gaussian noise are then added to these point to simulate the inherent elevation accuracy  $\sigma_s$  from the TomoSAR processing, with  $\sigma_s$  being 1 and 5 m, respectively. Therefore, total 10 Monte Carlo simulations of 10 000 realizations each were performed. The simulation result is shown in Fig. 15, which shows the end point accuracy improves with respect to increasing point density. The end point location accuracy  $\sigma_{\text{point}}$  is below 30 cm in the façade direction for a typical five-story high building, i.e., with façade point density being about 15/m<sup>2</sup> in the case

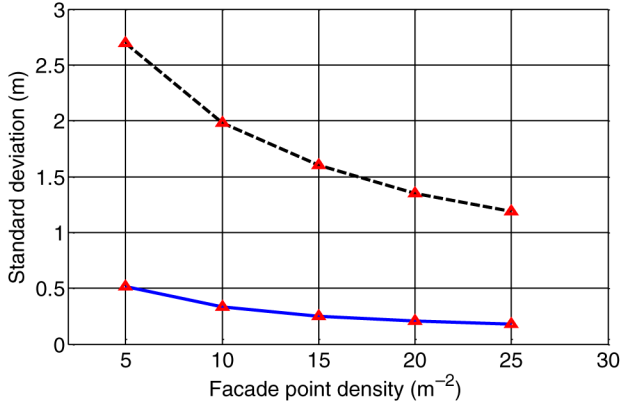


Fig. 15. Standard deviation of the end point estimates versus the façade point density. The non-façade point density is kept at  $1/\text{m}^2$ . The black-dashed line corresponds to a point cloud accuracy of 5 m in the façade direction, and the blue solid line corresponds to the accuracy of 1 m.

of meter resolution SAR data. The estimation accuracy also improves as the façade point density increases. For a typical urban area, the façade point density ranges from 5 to  $25/\text{m}^2$ , which corresponds to an accuracy of 50–20 cm. The final fusion accuracy will be typically varying from 5 to 30 cm, if we set  $N_{\text{pairs}}$  to be 100.

One should be aware of that the 3-D coordinates of a real TomoSAR point cloud are always discretized, due to the range, azimuth, and elevation sampling of 3-D SAR imaging. Therefore, the discretization in the façade direction also depends on the same facts, but in addition is subject to the heading angle of the satellite and the direction of the façade which is random. Consequently, the discretization along the façade direction is different for each façade. A simulation is performed for a discretization level of 0.5 m using the same aforementioned setting. No significant degradation of estimation accuracy is found.

### B. Parameters Settings

The proposed algorithm involves several parameters setting, including the point cloud filtering, and the weighted Hough transform for L-shape detection. Although these parameters seems to be data dependent, they are actually very much according to the inherent scale of the urban structures, e.g., the street width, number of building floors, minimum building size, etc. They can be reasonably derived with some prior knowledge of the urban structure. In the following, we give the parameter setting for TomoSAR point clouds derived from TerraSAR-X high-resolution spotlight data.

- 1) *Point cloud filtering*: we found using 20–50 nearest points, and a threshold at 10–20 m are suitable.
- 2) *2-D cross correlation*: the sampling distance of raster image should be less than half of the typical street width. We use 3 m in our algorithm.
- 3) *Point density estimation*: linear filter length is 10 m, which is a reasonable value for a shortest façade length, and width is set to be close to data resolution, i.e., 1 m.
- 4) *L-shape detection*:  $l_{\min}$  is set to 10 m, which is the shortest façade length, and  $d_{\min}$  is set to  $2/\text{m}^2$ , which



(a)



(b)

Fig. 16. (a) Google optical image of the test area and (b) the incoherent average SAR amplitude of the test area.

corresponds to the point density of a one to two-story high building in our TomoSAR point cloud. This leads to  $H_{\min} = 40$  ( $\Delta r = 1\text{m}$ ).

- 5) *Façade end points position estimation*: the rectangle filter size to be 5 m that corresponds to half of the shortest façade length.

## IV. APPLICATION ON REAL DATA

### A. Dataset

The proposed algorithm was tested on two TomoSAR point clouds generated from an ascending and a descending stack of high resolution TerraSAR-X spotlight data of Berlin. The ascending stack comprised of 79 interferograms and the descending stack has 94. The InSAR stacking and TomoSAR processing were done by the *PSI-GENESIS* [8], [51] and *Tomo-GENESIS* [18] systems, the PSI and TomoSAR processing system of Remote Sensing Technology Institute of DLR. Fig. 16 is the optical image from Google Earth, and an incoherent average of SAR amplitude of the ascending stack.

### B. Fusion Result

Each point cloud contains about 20 million points. Around 500 L-shapes are detected from each point cloud, corresponding to 1000 end points. Fig. 17(a) and (b) shows the detected L-shapes and end points of the two point clouds, overlaying on their gray-scale point density images, respectively. The matched L-shapes end points are shown in Fig. 17(c).





Fig. 17. (a) and (b) shows the detected L-shapes and their end points of the two input point clouds over Berlin and (c) the matched L-shapes and end points. The L-shapes are plotted in green, and the end points are marked as red. They are overlaid on the gray scale point density image.

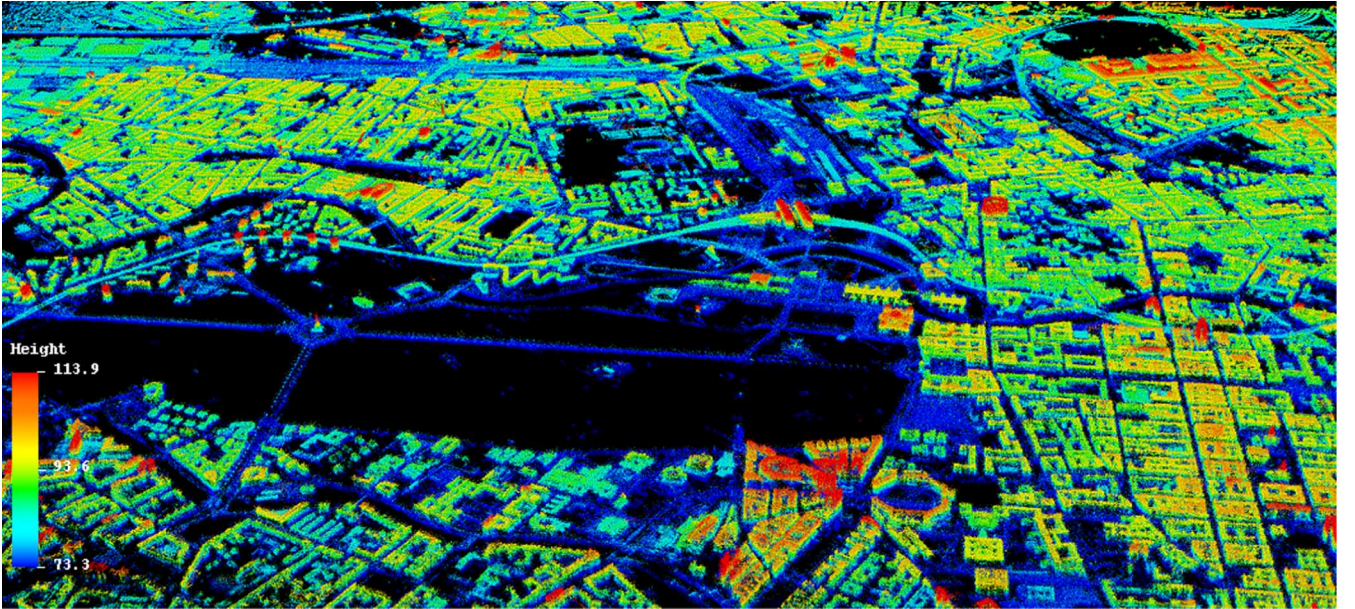


Fig. 18. Fusion result of ascending and descending point cloud of Berlin. The color indicates height on WGS84 reference surface. The unit of the color bar is meter.

Closed quadrilateral implies both pairs of end points are identified, a single L-shape implies only one pair of end points is identified. In total, 150 pairs of end points are matched, which is sufficient to robustly estimate the two unknowns  $\Delta z^a$  and  $\Delta z^b$ .

This result is compared with the result computed using the method explained in [1]. A difference of 0.51 m in *east* direction; 0.09 m in *north* direction, and almost no difference in height is found. A 3-D view of the complete fusion result is shown in Fig. 18. The height of the points is color-coded. The fused point cloud provides a Google-Street-View like visual sensation of the study area with very high level of detail. Such shadow-free TomoSAR point clouds are very useful for dynamic city model reconstruction [52] and scientific visualization [53], [54].

## V. CONCLUSION AND OUTLOOK

This paper introduced a robust method to fuse two TomoSAR point clouds derived from meter-resolution SAR data stacks of cross-heading orbits, i.e., ascending and descending orbits. The fusion is done by finding the L-shaped buildings from the two point cloud, and matching the end points of the two L-shapes belong to the same building.

The proposed algorithm was successfully tested on TomoSAR point clouds obtained from two stacks of high-resolution TerraSAR-X spotlight data. It is also compared with the fusion result using the only existing method [1] for fusing PSI point clouds derived from meter-resolution SAR data stacks of urban areas. Submeter consistency is achieved. Compared with [1], the computational speed has been greatly improved,



due to the reduced point cloud in the final RANSAC matching step. The fusion accuracy is theoretically better, since the matched pairs of points are theoretically the exact correspondence. The simulation shows the estimation accuracy of façade end point ranges from 20 cm to a few meters, depends on the façade point density (i.e., building height) and the quality of the TomoSAR point cloud itself.

The proposed method can be applied to point cloud with point density up to 10 times lower than the TerraSAR-X high resolution data, e.g., stripmap data. In order to extract sufficient number of buildings, instead of setting a threshold on point density only, a soft threshold should be set first, and then use surface normal information of the points to further classify them. This method is explained and successfully applied in [55].

The proposed algorithm requires a point cloud segmentation step, which requires either a sophisticated method or external data source, so the authors employ GIS data for assistance. The final fusion accuracy also depends on the abundance of quadrilateral buildings in the scene. Therefore, to improve the proposed method, the following work is already undergoing: 1) automatic point cloud segmentation and 2) modeling building façades using piecewise line segment instead of using L-shape only.

#### ACKNOWLEDGMENT

The authors would like to thank L. Ding from TUM-LFK for providing the source of Berlin building shape layer.

#### REFERENCES

- [1] S. Gernhardt and R. Bamler, "Deformation monitoring of single buildings using meter-resolution SAR data in PSI," *ISPRS J. Photogramm. Remote Sens.*, vol. 73, pp. 68–79, Sep. 2012.
- [2] G. Fornaro, F. Serafino, and F. Soldovieri, "Three-dimensional focusing with multipass SAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 3, pp. 507–517, Mar. 2003.
- [3] F. Lombardini, "Differential tomography: A new framework for SAR interferometry," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 1, pp. 37–44, Jan. 2005.
- [4] G. Fornaro, D. Reale, and F. Serafino, "Four-dimensional SAR imaging for height estimation and monitoring of single and double scatterers," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 1, pp. 224–237, Jan. 2009.
- [5] X. Zhu and R. Bamler, "Very high resolution spaceborne SAR tomography in urban environment," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 12, pp. 4296–4308, Dec. 2010.
- [6] X. Zhu and R. Bamler, "Let's do the time warp: Multicomponent nonlinear motion estimation in differential SAR tomography," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 4, pp. 735–739, Jul. 2011.
- [7] A. Ferretti, C. Prati, and F. Rocca, "Permanent scatterers in SAR interferometry," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 1, pp. 8–20, Jan. 2001.
- [8] N. Adam, B. Kampes, M. Eineder, J. Worawattanamatekul, and M. Kircher, "The development of a scientific permanent scatterer system," in *Proc. ISPRS Workshop High Resolut. Mapping Space*, Hannover, Germany, 2003, vol. 2003, p. 6.
- [9] B. M. Kampes, *Radar Interferometry—Persistent Scatterer Technique*, vol. 12. Dordrecht, The Netherlands: Springer, 2006.
- [10] S. Gernhardt, N. Adam, M. Eineder, and R. Bamler, "Potential of very high resolution SAR for persistent scatterer interferometry in urban areas," *Ann. GIS*, vol. 16, no. 2010–06, pp. 103–111, 2010.
- [11] P. Berardino, G. Fornaro, R. Lanari, and E. Sansosti, "A new algorithm for surface deformation monitoring based on small baseline differential SAR interferograms," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 11, pp. 2375–2383, Nov. 2002.
- [12] R. Lanari *et al.*, "A small-baseline approach for investigating deformations on full-resolution differential SAR interferograms," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 7, pp. 1377–1386, Jul. 2004.
- [13] K. Goel and N. Adam, "An advanced algorithm for deformation estimation in non-urban areas," *ISPRS J. Photogramm. Remote Sens.*, vol. 73, pp. 100–110, Sep. 2012.
- [14] A. Ferretti *et al.*, "A new algorithm for processing interferometric data-stacks: SqueeSAR," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 9, pp. 3460–3470, Sep. 2011.
- [15] Y. Wang, X. Zhu, and R. Bamler, "Retrieval of phase history parameters from distributed scatterers in urban areas using very high resolution SAR data," *ISPRS J. Photogramm. Remote Sens.*, vol. 73, pp. 89–99, Sep. 2012.
- [16] G. Fornaro, A. Paucillo, D. Reale, and S. Verde, "Improving SAR tomography urban area imaging and monitoring with CAESAR," in *Proc. 10th Eur. Conf. Synth. Aperture Radar (EUSAR'14)*, 2014, pp. 1–4.
- [17] X. Zhu, *Very High Resolution Tomographic SAR Inversion for Urban Infrastructure Monitoring—A Sparse and Nonlinear Tour*. Munich, Germany: Deutsche Geodätische Kommission, Verlag der Bayerischen Akademie der Wissenschaften, 2011.
- [18] X. Zhu, Y. Wang, S. Gernhardt, and R. Bamler, "Tomo-GENESIS: DLR's Tomographic SAR processing system," in *Proc. Joint Urban Remote Sens. Event (JURSE)*, 2013, pp. 159–162.
- [19] X. Zhu and R. Bamler, "Tomographic SAR inversion by L1-norm regularization—The compressive sensing approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 10, pp. 3839–3846, Oct. 2010.
- [20] X. Zhu and R. Bamler, "Demonstration of super-resolution for tomographic SAR imaging in urban environment," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 8, pp. 3150–3157, Aug. 2012.
- [21] Y. Wang, X. Zhu, and R. Bamler, "An efficient tomographic inversion approach for urban mapping using meter resolution SAR image stacks," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 7, pp. 1–5, Jul. 2014.
- [22] Data provided by "Land Berlin" and "Business Location Service," supported by "Europäischer Fonds für Regionale Entwicklung."
- [23] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, 1999, vol. 2, pp. 1150–1157.
- [24] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up Robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, Jun. 2008.
- [25] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR'05)*, 2005, vol. 1, pp. 886–893.
- [26] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA'09)*, 2009, pp. 3212–3217.
- [27] S. Ruiz-Correa, L. G. Shapiro, and M. Melia, "A new signature-based method for efficient 3-D object recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR'01)*, 2001, vol. 1, pp. 1769–1776.
- [28] B. Steder, R. B. Rusu, K. Konolige, and W. Burgard, "Point feature extraction on 3D range scans taking into account object boundaries," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2011, pp. 2601–2608.
- [29] L. Silva, O. R. P. Bellon, and K. L. Boyer, "Precision range image registration using a robust surface interpenetration measure and enhanced genetic algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 762–776, May 2005.
- [30] O. Cordón, S. Damas, and J. Santamaría, "A fast and accurate approach for 3D image registration using the scatter search evolutionary algorithm," *Pattern Recognit. Lett.*, vol. 27, no. 11, pp. 1191–1200, Aug. 2006.
- [31] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.
- [32] Z. Zhang, "Iterative point matching for registration of free-form curves and surfaces," *Int. J. Comput. Vis.*, vol. 13, no. 2, pp. 119–152, 1994.
- [33] G. C. Sharp, S.-W. Lee, and D. K. Wehe, "ICP registration using invariant features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 1, pp. 90–102, Jan. 2002.
- [34] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann, "Robust global registration," in *Proc. Symp. Geom. Process.*, 2005, vol. 2, p. 5.
- [35] A. Makadia, A. I. Patterson, and K. Daniilidis, "Fully automatic registration of 3D point clouds," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Washington, DC, USA, 2006, vol. 1, pp. 1297–1304.
- [36] A. W. Fitzgibbon, "Robust registration of 2D and 3D point sets," *Image Vis. Comput.*, vol. 21, no. 13–14, pp. 1145–1153, Dec. 2003.



- [37] A. Gruen and D. Akca, "Least squares 3D surface and curve matching," *ISPRS J. Photogramm. Remote Sens.*, vol. 59, no. 3, pp. 151–174, May 2005.
- [38] P. Gamba, F. Dell'acqua, and B. Houshmand, "Comparison and fusion of LIDAR and InSAR digital elevation models over urban areas," *Int. J. Remote Sens.*, vol. 24, no. 22, pp. 4289–4300, 2003.
- [39] L. Bornaz, A. Lingua, and F. Rinaudo, "Multiple scan registration in LIDAR close range applications," *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.*, vol. 34, pp. 72–77, 2003.
- [40] W. Von Hansen, H. Gross, and U. Thoennessen, "Line-based registration of terrestrial and airborne LIDAR data," *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.*, vol. 37, pp. 161–166, 2008.
- [41] J.-J. Jaw and T.-Y. Chuang, "Registration of ground-based LiDAR point clouds by means of 3D line features," *J. Chin. Inst. Eng.*, vol. 31, no. 6, pp. 1031–1045, 2008.
- [42] A. Gressin, C. Mallet, J. Demantké, and N. David, "Towards 3D lidar point cloud registration improvement using optimal neighborhood knowledge," *ISPRS J. Photogramm. Remote Sens.*, vol. 79, pp. 240–251, 2013.
- [43] Y. Wang, X. Zhu, Y. Shi, and R. Bamler, "Operational TomoSAR processing using TerraSAR-X high resolution spotlight stacks from multiple view angles," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, 2012, pp. 7047–7050.
- [44] E. Simonetto, H. Oriot, and R. Garelo, "Rectangular building extraction from stereoscopic airborne Radar images," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 10, pp. 2386–2395, Oct. 2005.
- [45] F. Zhang, Y. Shao, X. Zhang, and T. Balz, "Building L-shape footprint extraction from high resolution SAR image," in *Proc. Joint Urban Remote Sens. Event (JURSE)*, 2011, pp. 273–276.
- [46] F. Zhang, L. Liu, and Y. Shao, "Building footprint extraction using dual-aspect high-resolution synthetic aperture radar images in urban areas," *J. Appl. Remote Sens.*, vol. 6, no. 1, pp. 063599–063599, 2012.
- [47] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.
- [48] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, pp. 11:1–11:37, Jun. 2011.
- [49] OpenStreetMap shape layer provided for free by Geofabrik GmbH [Online]. Available: <http://download.geofabrik.de/europe/germany>, accessed on Oct. 29, 2012.
- [50] S. Gernhardt, X. Cong, M. Eineder, S. Hinz, and R. Bamler, "Geometrical fusion of multitrack PS point clouds," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 1, pp. 38–42, Jan. 2012.
- [51] N. Adam, M. Eineder, N. Yague-Martinez, and R. Bamler, "High resolution interferometric stacking with TerraSAR-X," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS'08)*, 2008, vol. 2, pp. II-117–II-120.
- [52] X. Zhu and M. Shahzad, "Façade reconstruction using multiview spaceborne TomoSAR point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 6, pp. 3541–3552, Jun. 2014.
- [53] L. Ding, X. Zhu, and L. Meng, "Visual analysis of large amounts of 4-D building deformation data," in *Proc. 26th Int. Cartographic Conf.* (ISBN: 978-1-907075-06-3), Dresden, Germany, 2013, pp. 287–297.
- [54] D. Liang, T. Balz, Z. Wang, L. Wei, and M. Liao, "Web-based interactive visualization of PS-InSAR point clouds for education and training," *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.*, vol. II-6, pp. 7–12, Apr. 2014. Demo website [Online]. Available: <http://sarviz.org/Psi/View/LasVegas>
- [55] M. Shahzad and X. Zhu, "Robust reconstruction of building facades for large areas using spaceborne TomoSAR point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 752–769, Feb. 2015.



**Yuanyuan Wang** (S'11) received his B.Eng. (Hons.) degree in electrical engineering from The Hong Kong Polytechnic University, Hung Hom, Hong Kong, in 2008, and the M.Sc. degree in Earth Oriented Space Science and Technology (ESPACE) from Technische Universität München (TUM), München, Germany, in 2010. Since January 2011, he has been pursuing the Doctoral degree with the TUM-IGSSE research team "4D City."

Currently, he is with the Helmholtz Young Investigators Group "SiPEO." In June and July of 2014, he was a Guest Scientist at the Institute of Visual Computing, ETH Zürich, Switzerland. His research interests include optimal parameters estimation in InSAR techniques, multisensor fusion algorithms of SAR/optical data, and the applications of these techniques in urban and volcanic areas.



**Xiao Xiang Zhu** (S'10–M'12–SM'14) received the Bachelor's degree in space engineering from the National University of Defense Technology (NUDT), Changsha, China, in 2006, the M.Sc., Dr.-Ing. degrees, and the "Habilitation" in the field of signal processing from Technische Universität München (TUM), Munich, Germany, in 2008, 2011, and 2013, respectively.

During October/November 2009, she was a Guest Scientist with the Italian National Research Council (CNR), Institute for Electromagnetic Sensing of the Environment (IREA), Naples, Italy. Since May 2011, she has been a Scientist with the Remote Sensing Technology Institute, German Aerospace Center (DLR), Oberpfaffenhofen, Weßling, Germany, where she is the Head of the Team Signal Analysis, and with the Chair of Remote Sensing Technology, TUM. Since September 2013, she has been leading the Helmholtz Young Investigators Group "SiPEO" and is appointed as TUM Junior Fellow. Her research interests include modern signal processing, including innovative algorithms such as compressive sensing and sparse reconstruction, with applications in the field of remote sensing such as multi/hyperspectral image analysis; advanced InSAR techniques such as high-dimensional tomographic SAR imaging and SqueeSAR; and computer vision in remote sensing including object reconstruction and multidimensional data visualization.