Dense Surface Models from Airborne and Spaceborne (Multi-)Stereo Images

Thomas Krauß

Abstract Digital surface models representing the heights of an area can be derived from two or more (multi-)stereo images of airborne or spaceborne sensors. A satellite stereo image of a current very-high-resolution satellite like WorldView or GeoEye with ground pixel sizes of about half a meter allows the derivation of surface models in the range of the same resolution. Such surface models are the basis of many applications like the three-dimensional representation of the area, 3D change detection, calculation of volumes, detection of sight lines, or water flow and flooding. Satellite imagery covers large areas of about 400 square kilometers with ground resolution of about 1 meter, while airborne images from planes or drones usually cover only small areas but with higher resolution. In this chapter the basics of digital surface models are shown, and the actually best method for deriving dense digital surface models from airborne and spaceborne images is described. Some examples finally show the possible results.

Keywords Digital surface models · Dense stereo matching · Satellite-borne stereo imagery · Airborne stereo imagery · Semi-global matching

What Are Dense Digital Surface Models?

Digital surface models represent heights of an area on Earth's surface. For this a digital surface model – subsequently referred to as DSM – is a georeferenced image containing a height value, e.g. above sea level, for each point of the area. The DSM represents the upper boundary of all objects on Earth as the heights of roofs or the top of trees or other vegetation areas. In contrast the digital terrain model (DTM) represents the height of the bare ground without any natural or man-made objects on it. The latter will be of interest, e.g. in the calculation of water runoff, and can be derived from a dense DSM.

T. Krauß (✉)
German Aerospace Center (DLR), Remote Sensing Technology Institute, Weßling, Germany
e-mail: thomas.krauss@dlr.de

D. G. Hadjimitsis et al. (eds.), *Remote Sensing for Archaeology and Cultural Landscapes*, Springer Remote Sensing/Photogrammetry,
https://doi.org/10.1007/978-3-030-10979-0_6

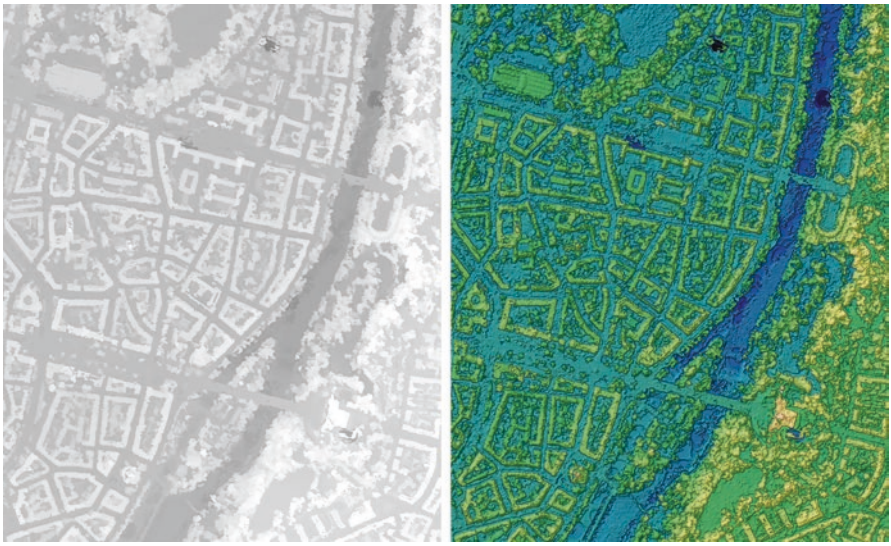

Fig. 1 Example of a dense DSM of the center of Munich (1.1 × 1.35 square km); left, DSM representation as gray values (black = 500 m, white = 600 m above WGS84 ellipsoid); right, shaded and color-coded DSM

A "dense" digital surface model is a DSM with individually calculated heights for each possible pixel of the input images in contrast to DSMs interpolated from only a few derived height values. Figure 1 shows a dense DSM derived from WorldView-2 satellite data acquired 10th of July 2012 over the center of Munich (Germany). The DSM is generated at a ground sampling distance (pixel size) of 0.5 m which is also the resolution of the satellite images. In Fig. 1, left, the representation of the DSM in gray values is shown. Lighter areas are higher than dark areas. The gray coding in this DSM runs from 500 m above WGS84 ellipsoid (black) to 600 m (white). Figure 1, right, shows the same DSM in a color-coded and shaded representation which is more intuitive for most people.

Digital surface models can be extracted from any kind of stereo imagery acquired from above the area in question. Having two images of the same area from different viewpoints allows the calculation of relative parallaxes (so-called disparities) between points in the images. By using the known acquisition position, viewing angles and internal parameters of the camera also the absolute height as shown in Fig. 2 can be derived.

The two acquired images – referred as "left" and "right" image in Fig. 2 – show objects of different heights at different relative positions. The relative difference or disparity allows the calculation of the absolute height if the position and camera parameters of the sensor are known.

Airborne images from planes or drones are mostly acquired as frame camera images – the whole image is acquired at once. To take stereo images, only an

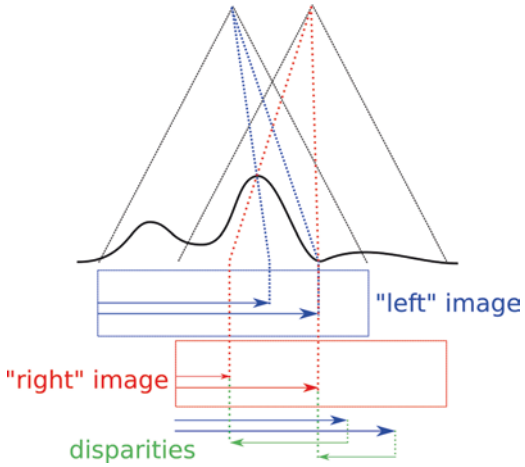

Fig. 2 Principle of the derivation of heights from stereo imagery – higher points (nearer to the sensor) show larger disparities

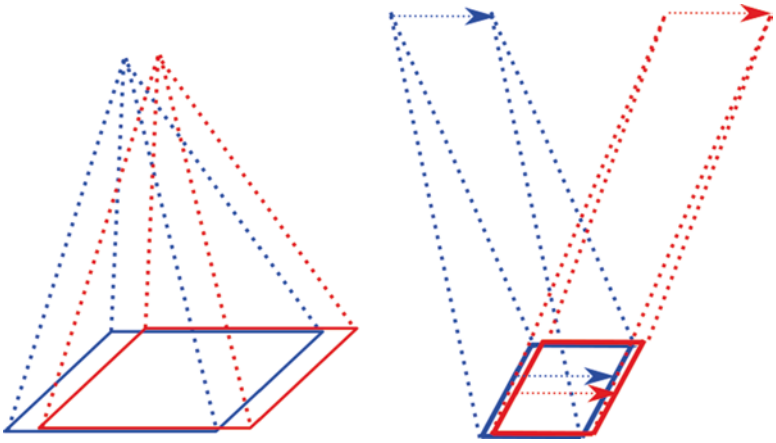

Fig. 3 Frame camera (left) vs. push broom sensor (right)

in-flight-direction acquisition of two or more images with high overlaps is required. In contrast satellite-borne images are mostly acquired by so-called push broom sensors. A push broom sensor has only one line of CCD pixels and builds up the image while scanning over an area line by line during the movement of the satellite in orbit. To fetch the second stereo image, the sensor has to be rotated, and the area under investigation has to be scanned a second time as shown in Fig. 3.

Image Correlation

To derive a DSM from two (or more) stereo images, it is therefore needed to find identical points in the images. Since the (automatic) generation of surface models has already a long history in image processing and Earth observation, there exists a broad amount of approaches. Here the classical approach will be presented together with a simple dense matching and the actually most advanced method – the semi-global matching (SGM).

The classical approach searches for corresponding corner points in the two stereo images using an image pyramid approach with a window-based cross-correlation matching and local least squares sub-pixel refinement.

In contrast the simple dense matching method utilizes the fact that two stereo images can be transformed to a so-called epipolar geometry in which the disparities are only horizontal shifts and no more general two-dimensional vectors in the images. But this simple method suffers from the missing linkage between subsequent lines, so the resulting dense DSMs show many stripes or streaking effects (Scharstein and Szeliski 2002).

A solution for this is the semi-global matching developed in 2005 at DLR (Hirschmüller 2005). This method interconnects the disparities of the epipolar lines using eight or sixteen scanning directions for calculation.

A benchmark initiated by Scharstein and Szeliski (2002) collected for long years all relevant approaches from computer vision for derivation of disparity maps from stereo imagery. For a long time, graph-based methods like "maximum flow minimum cut" competed with dynamic programming-based methods like described here. Nowadays the semi-global matching (Hirschmüller 2005) together with its improvements for satellite imagery (d'Angelo et al. 2008) is actually the fastest and qualitatively best possible operational approach.

Classical Approach

The classic approach (Lehner and Gill 1992; Otto and Chau 1989) is based on the detection of so-called interest points in one of the images and finds the corresponding points in the other stereo mate. These interest points are corner points with strong gradients in two perpendicular directions.

First the stereo images are scaled down in an image pyramid by 1/2, 1/4, …, $1/2^n$ (e.g., 1/64). In the smallest image, interest points are searched. These points are correlated to the corresponding points in the stereo partner image by using a window-based cross-correlation matching followed by a sub-pixel local least squares matching. The found correlations are propagated to the next higher level of the pyramid as first estimations of the correlating window positions for the matching, and the steps "search interest points," "window based cross correlation," and "local least squares matching" are applied to this pyramid level.

In the final level – the original stereo images – the detected correlations are densified using a region growing approach. Finally the result is a list of pixel coordinates in the first stereo image together with the sub-pixel positions in the second stereo image – their distances representing the disparities.

Using the sensor model, real 3D coordinates can be derived for every correlated point, and all these points can be interpolated to a final DSM as shown in Fig. 4, left.

Epipolar Geometry

As can be seen in Fig. 4, left, the classical approach is not suitable in urban areas. Since there remain only a few good "corner points" and these get correlated by a window with a size larger than one pixel, the result is always a smooth, interpolated DSM. In contrast Fig. 4, right, shows a DSM derived from the same images using a more sophisticated "dense" method.

In the next approach, the fact is used that any image stereo pair can be transformed to a so-called epipolar geometry in which the disparities are reduced to simple shifts in only row or column direction. We use here – without losing generality – the horizontal epipolar direction.

Figure 5 shows the principle of creating epipolar images. Two stereo images left and right represented in blue, respectively, and red together with their projection centers Z_L and Z_R see the same object point P in the left image in P_L and in the right image in P_R. Moving the object point to P′ without changing P_R moves the projection P_L to P'_L I in the left image. The line $P_L P'_L$ represents now the epipolar line in the left image. Doing the same with object point Q and the right image gives the epipolar line $Q_R Q'_R$ in the right image.

Now both original stereo images can be rotated around their center in a way they found epipolar lines $P_L P'_L$ and $Q_R Q'_R$ get both horizontally. Figure 6 shows again sections of the two stereo images of the center of Munich, and Fig. 7 shows the corresponding epipolar images.

Calculating the disparities now in the epipolar images requires at end of course the back projection to the original images for applying the sensor model and in turn calculating the real absolute ellipsoid heights.

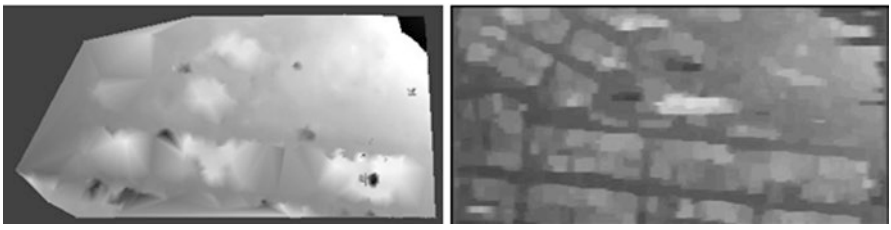

Fig. 4 Small section 400 m × 200 m from an Ikonos-Scene of Athens (acquired 2004), left, DSM derived using the classical approach; right, DSM derived using a simple dense stereo approach

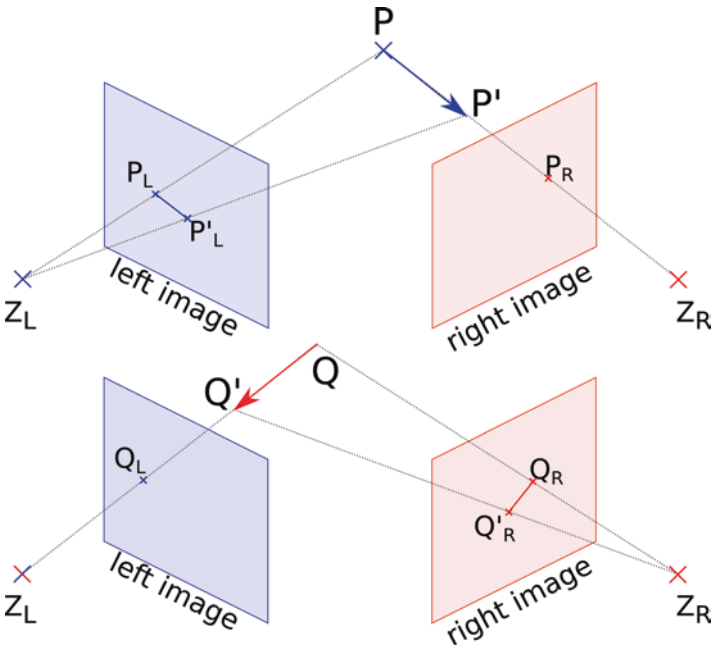

Fig. 5 Construction of epipolar geometry

Fig. 6 Original stereo images (750 m × 750 m, Center of Munich, Germany, 10 July, 2012, Worldview-2 (2012, European Space Imaging)

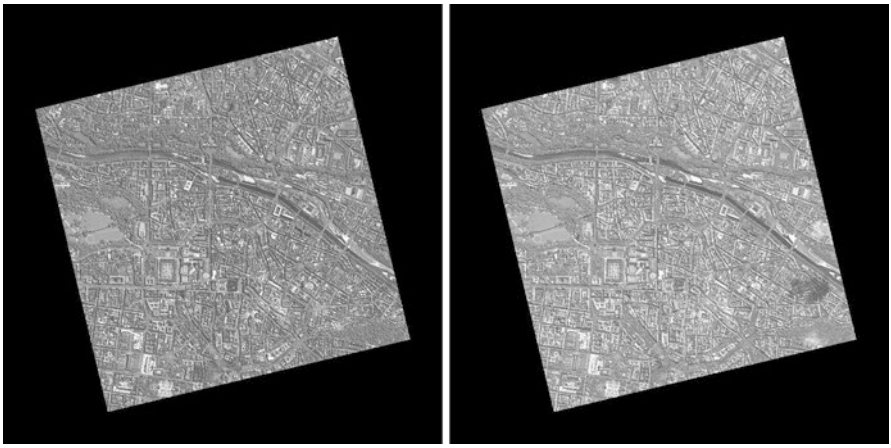

Fig. 7 Epipolar images transformed from Fig. 6

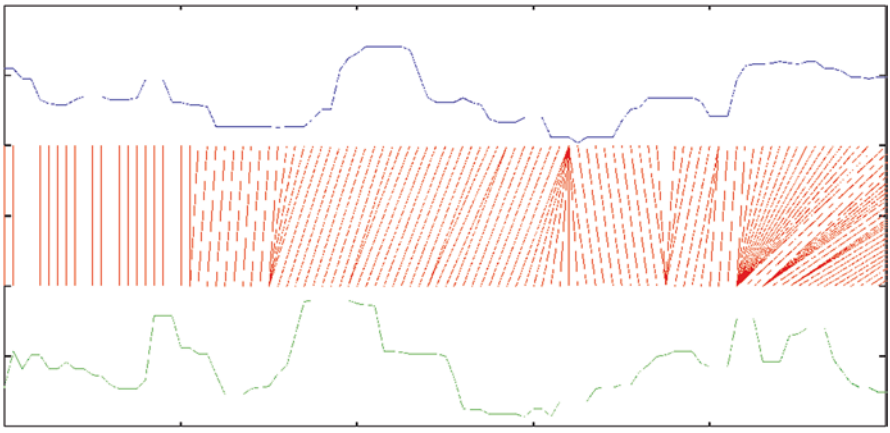

Fig. 8 Example of two gray value profiles of corresponding epipolar lines from two stereo images (top and bottom) together with the required shifts for warping one onto the other (middle)

Dynamic Programming

Using the epipolar geometry allows the reduction of a 2D correlation problem to a much simpler 1D problem by warping each epipolar line of one stereo image onto the corresponding line of the second epipolar stereo image (see Fig. 8).

A solution for the 1D correlation problem is applying the "dynamic programming" as described in (Birchfield and Tomasi 1998; Krauß et al. 2005; van Meerbergen et al. 2002). For finding the best local shifts between the epipolar lines in a $m \times n$-matrix (m and n are the lengths of the epipolar lines in the left and right

stereo image) so-called costs C between pixel i of the epipolar line of image I_L and j of the epipolar line of image I_R will be stored in $C(i,j)$. The cost can be as simple as the absolute difference of the gray values of the image pixels.

Now a path from (0, 0) to (m, n) in the matrix has to be found which minimizes the sum of all $C(i,j)$ on this path. The distances on this path to the main diagonal give the local shifts at this position. A recursive solution may look like

```
function getmin(i,j):
  minpos := None
  minc := 0
  (crek, posrek) := (0, [])
  for k in [max(0,j − d) . . . min(j + d,len(I^I)−1]:
    if i <len(I)−1: (crek, posrek) := getmin(i + 1, k + 1)
    c = |I(i) − I^I (k)| + crek
    if minpos is None or c < minc:
       minpos := k
       minc := c
 return (minc, [minpos] + posrek) (totalmindist, minpath) = getmin(1,1)
(totalmindist, minpath) = getmin(1,1)
```

But a more sophisticated and easier solution gives the "dynamic programming." In this method the cost matrix C is summarized in a special way up to an aggregated matrix D. $D(m, n)$ now contains the absolute minimum distance, and going back from (m, n) to (0, 0) just by taking always the minimum upper or left neighbor gives the searched minimum path or the searched shifts between the two epipolar lines. D is derived from C by setting the first line and column to

$$D_{1,j} = \sum_{k=1}^{j} C_{1,k}, \quad D_{i,1} = \sum_{k=1}^{i} C_{k,1} \tag{1}$$

and the remaining $D_{i,j}$ ($i, j > 1$) to

$$D_{i,j} = C_{i,j} + \min\left(D_{i-1,j}, D_{i,j-1}, D_{i-1,j-i}\right) \tag{2}$$

The dynamic programming approach can be illustrated best using a simple example. Given are two gray value arrays ("epipolar lines" of the two stereo images) I and I':

$$I = \begin{pmatrix} 1 \\ 0 \\ 2 \\ 1 \\ 0 \end{pmatrix} \quad \text{und} \quad I' = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 2 \\ 1 \end{pmatrix} \tag{3}$$

From them the cost matrix C will be calculated to $C_{i,j} = |I_i - I'_j|$:

$$C = \begin{vmatrix} 1-0 & 1-1 & 1-0 & 1-2 & 1-1 \\ 0-0 & 0-1 & 0-0 & 0-2 & 0-1 \\ 2-0 & 2-1 & 2-0 & 2-2 & 2-1 \\ 1-0 & 1-1 & 1-0 & 1-2 & 1-1 \\ 0-0 & 0-1 & 0-0 & 0-2 & 0-1 \end{vmatrix} = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 2 & 1 \\ 2 & 1 & 2 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 2 & 1 \end{pmatrix} \tag{4}$$

Aggregation of C following Eqs. 1 and 2 gives

$$D = \begin{pmatrix} \mathbf{1} & \mathbf{1} & 2 & 3 & 3 \\ 1 & 2 & \mathbf{1} & 3 & 4 \\ 3 & 2 & 3 & \mathbf{1} & 2 \\ 4 & 2 & 3 & 2 & \mathbf{1} \\ 4 & 3 & 2 & 4 & \mathbf{2} \end{pmatrix} \tag{5}$$

with the minimum path marked in bold in the latter. The total minimum distance is always the rightmost bottom element – in our case 2. Starting from there using always the smallest neighbor left, top, or top left results in the minimum path. The distance of the minimum path from the main diagonal gives the disparity d as

$$d = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{6}$$

There are many image processing approaches for connecting the individually calculated disparities of each line. So it's possible to use small windows of diameter w instead of only one pixel for cost calculation like in

$$C_{i,j} = \sum_{\lambda=-w/2}^{w-w/2-1} \sum_{\mu=-w/2}^{w-w/2-1} \left| I_{i+\lambda,y+\mu} - I'_{j+\lambda,y+\mu} \right| \tag{7}$$

Also applying a median filtering with diameter m after the disparity calculation will help smoothing the result. Figure 9 shows four examples calculated with different window sizes w and median sizes m. But using such filtering approaches – especially window sizes w larger than one pixel – will result always in a smoothing of the resulting DSM in the same size, which is mostly not applicable particularly in reconstruction of building structures.

This leads to the actually best dynamic programming interconnection approach, the semi-global matching.

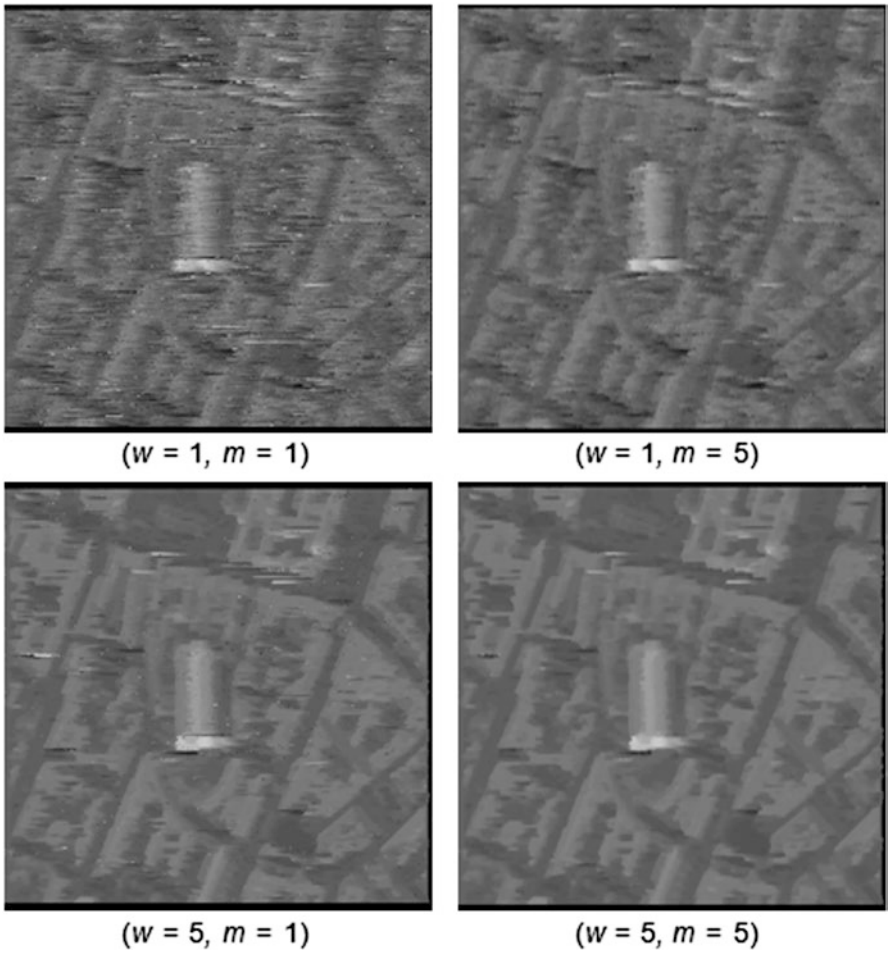

Fig. 9 DSMs calculated for given window sizes w and median sizes m (center of Munich, Frauenkirche 500 m × 500 m)

Semi-global Matching (SGM)

The semi-global matching (Hirschmüller 2005) – SGM below – extends the simple dynamic programming approach described above by interconnecting the disparities of the epipolar lines already in the cost aggregation step and cutting down the matrices C and D to only a small band with size $\pm d_{\max}$ around the main diagonal using an estimated maximum disparity $d_{\max}$.

But this approach increases the memory needed for the calculation since now not only two matrices C and D of size w_I^2 are needed but three- dimensional matrices of size $w_I \times h_I \times (2d_{\max} + 1)$ using w_I and h_I as width and height of the epipolar images (Fig. 10).

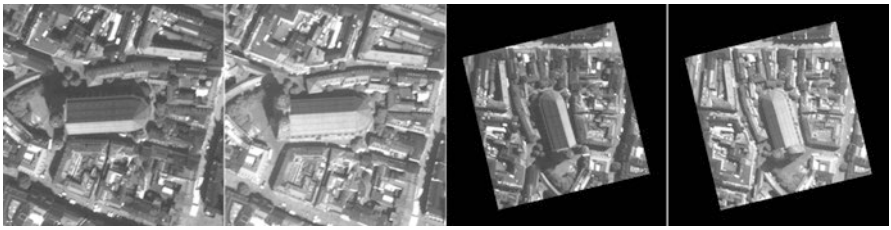

Fig. 10 Section 500 m × 500 m around Frauenkirche, center of Munich, Germany, left, original images; right, epipolar images

In the first step, the cost matrix is calculated again for horizontal-oriented epipolar images I_1 and I_2 using any kind of cost function like

$$C(x,y,d) = \left| I_1(x,y) - I_2(x+d,y) \right| \tag{8}$$

Another – in urban areas or noisy images very useful – type of cost function is the "census" function. Census (Spangenberg et al. 2013; Zabih and Woodfill 1994) does not use differences of gray values but only the relative brightnesses in respect to the center position. For this a census value $c_i(x, y)$ is calculated in image I_i as

$$c_i(x,y) = \bigotimes_{h_c/2}^{\mu=-h_c/2} \bigotimes_{w_c/2}^{\lambda=-w_c/2} s\left(I_i(x,y), I_i(x+\lambda, y+\mu)\right) \tag{9}$$

with

$$s(u,v) = \begin{cases} 0 \text{ for } u \le v \\ 1 \text{ for } u > v \end{cases} \tag{10}$$

using a small window of size $wc \times hc$ around each point (x, y) and ⊗ as the bit concatenation operator. So a census window of size 3 × 5 will result in a 15 bit value or a more usual 7 × 9 in a 63 bit value. The cost for calculating the cost matrix is now the Hamming distance $\Delta_H(u, v)$ – how many bits differ in the bit strings u and v. So the cost matrix will calculate to

$$C(x,y,d) = \Delta_H\left(c_1(x,y), c_2(x+d,y)\right) \tag{11}$$

For aggregating C to D, not only one aggregation in epipolar line direction as shown in Eqs. 1 and 2 is used. Hirschmüller proposes in Hirschmüller (2005) using at least 8 or better 16 directions r for aggregation as shown in Fig. 11. Also he proposes the usage of two additional smoothness parameters p_1 and p_2 allowing for slanted and curved surfaces. The aggregated cost matrix D is now the sum of all aggregations from all directions:

$$D = \sum L_r \tag{12}$$

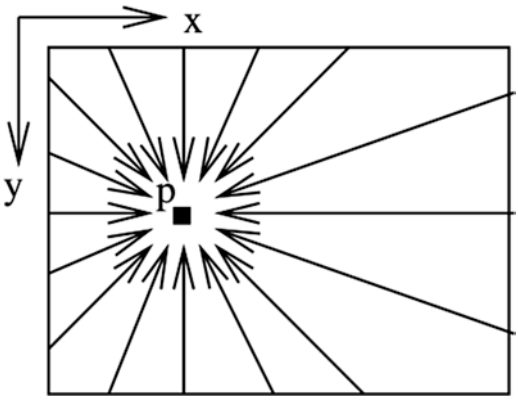

Fig. 11 SGM aggregation of 16 paths – each using again a dynamic programming 1D approach

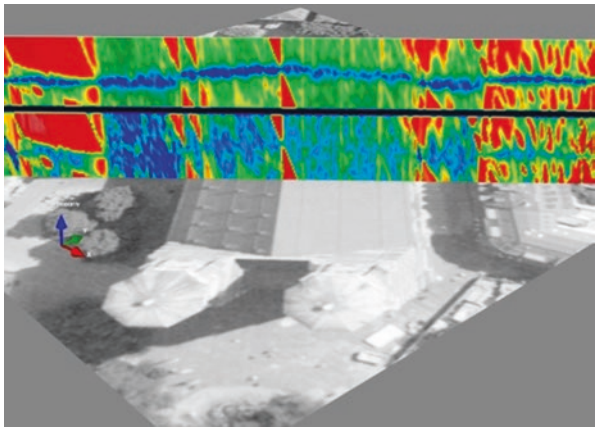

Fig. 12 Calculated cost C (bottom) and aggregated cost D (top) for one epipolar line, color-coded (blue to red from minimum to maximum cost)

with L_r being the aggregated cost matrix in direction r of $ndir$ directions ($ndir = 8$ or 16). The cost matrix L_r for direction $r = (r_x, r_y)$ is calculated as

$$\begin{aligned} L_{x,y,d} = C_{x,y,d} + \min(&L_{x-r_x,y-r_y,d}, \\ &L_{x-r_x,y-r_y,d-1} + p_1, \\ &L_{x-r_x,y-r_y,d+1} + p_1, \\ &\min_{i=-d_{\max}}^{d_{\max}} \left(L_{x-r_x,y-r_y,i}\right) + p_2) \end{aligned} \tag{13}$$

using the previously calculated disparity column $(x - r_x, y - r_y)$ and C. Figure 12 shows a section of the cost cube C and aggregated cost cube D for one aggregated epipolar line, disparities ranging from bottom to top from $-d_{\max}$ to $d_{\max}$.

In Fig. 13 the cost cube C and the aggregated cost cube D are shown for the example epipolar images from Fig. 10.

The calculation of the "minimum path" is now simply reduced to searching the disparity d of the minimum aggregated cost for each image pixel (x, y) in $D(x, y, d)$.

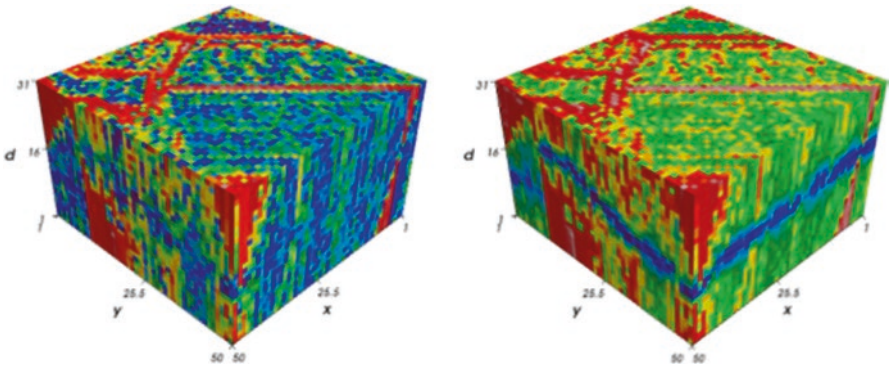

Fig. 13 Census cost cube C (left) and aggregated cost cube D (right) for the epipolar images from Fig. 10. (Images courtesy of Pablo d'Angelo)

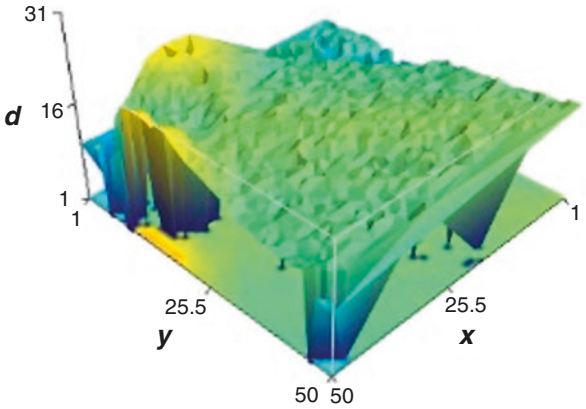

Fig. 14 Resulting disparity image P in the same 3D view as the cost cubes in Fig. 13. (Courtesy Pablo d'Angelo)

In Fig. 13, right, these are the disparities marked in blue. The result is the disparity image $P(x, y)$ as shown in Fig. 14.

For better results the parallaxes can be calculated first from I_1 to I_2 giving a disparity image P_1 fitting exactly on I_1 and second vice versa from I_2 to I_1 giving P_2 fitting exactly on I_2. Afterwards an outlier detection can be applied removing all disparities which differ by more than d_{lim} – usually about 1 px:

$$\Delta d(x,y) = \left| P_1(x,y) - \left(-P_2\left(x + P_1(x,y), y \right) \right) \right| < d_{\text{lim}} \tag{14}$$

The disparity at $P_1(x, y)$ tells us that the pixel (x, y) in the first epipolar image corresponds to pixel $(x + P_1(x, y), y)$ in the second image. Using this pixel in the second disparity map, P_2 should give the same but negative disparity if both are consistent.

Sensor Models: From Disparities to Absolute Heights

To calculate absolute heights from the disparities of the images, a so-called sensor model for the camera used is needed. Satellite imagery is mostly shipped with so-called RPCs (rational polynomial coefficients) (Grodecki et al. 2004; Jacobsen et al. 2005). These describe in a simplified third-order function the correlation of row (y) and column (x) of a pixel in the image with absolute longitude (λ) and latitude (φ) as a function of the absolute height h above the WGS84 ellipsoid:

$$x = \frac{f_{\text{samp,num}}(\lambda,\varphi,h)}{f_{\text{samp,den}}(\lambda,\varphi,h)} \quad \text{and} \quad y = \frac{f_{\text{line,num}}(\lambda,\varphi,h)}{f_{\text{line,den}}(\lambda,\varphi,h)} \tag{15}$$

or more generally as sensor model S_i for image i:

$$(x,y) = S_i(\lambda,\varphi,h) \tag{16}$$

Each of the functions $f()$ in Eq. 15 is a third-order polynomial in λ, φ, and h with 20 coefficients c_i = ‹*sl*›‹*nd*›_coeff_‹*i*› each with ‹*sl*› = {samp, line} ‹*nd*› = {num, den} and i ranging from 1 to 20. Together with 10 scale and offset parameters for x, y, λ, φ, and h, there are 90 parameters defining an RPC for a satellite image.

Each function $f_{\{\text{samp,line}\}-\{\text{num,den}\}}()$ is defined for its 20 coefficients c_i as

$$\begin{aligned} f(\lambda,\varphi,h) = {} & c_1 + c_2\lambda + c_3\varphi + c_4 h + c_5\lambda\varphi + c_6\lambda h + c_7\varphi h + \\ & c_8\lambda^2 + c_9\varphi^2 + c_{10}h^2 + c_{11}\lambda\varphi h + c_{12}\lambda^3 + c_{13}\lambda\varphi^2 + c_{14}\lambda h^2 + \\ & c_{15}\lambda^2\varphi + c_{16}\varphi^3 + c_{17}\varphi h^2 + c_{18}\lambda^2 h + c_{19}\varphi^2 h + c_{20}h^3 \end{aligned} \tag{17}$$

Using this sensor model allows the direct calculation of pixel coordinates x and y if the absolute geographic longitude λ and latitude φ are known together with the ellipsoidal height h. Calculating λ and φ vice versa from x and y for a known h needs an iterative solution until the required accuracy is reached (Fig. 15).

The disparity image P describes the correlation of pixels in the epipolar image I_1 with pixels in the epipolar image I_2. Let E_i be the epipolar transformation of image i and E_i^{-1} the inverted epipolar transformation of image i. For pixel (x, y) in the disparity image holds: pixel $I_1^{\text{orig}}(E_1^{-1}(x, y))$ is the same as $I_2^{\text{orig}}(E_2^{-1}(x + P(x, y), y))$, and for the sensor models, S_i can be stated

$$S_1(\lambda_1,\varphi_1,h) = E_1^{-1}(x,y) \quad \text{and} \quad S_2(\lambda_2,\varphi_2,h) = E_2^{-1}(x + P(x,y), y). \tag{18}$$

Varying the height h until $(\lambda_1, \varphi_1) = (\lambda\ 2, \varphi\ 2)$ provides the searched value of h for each point in I_1^{orig} creating a so-called "height map" M. This is an image fitting exactly on I_1^{orig} but containing the real absolute ellipsoidal heights h at each point.

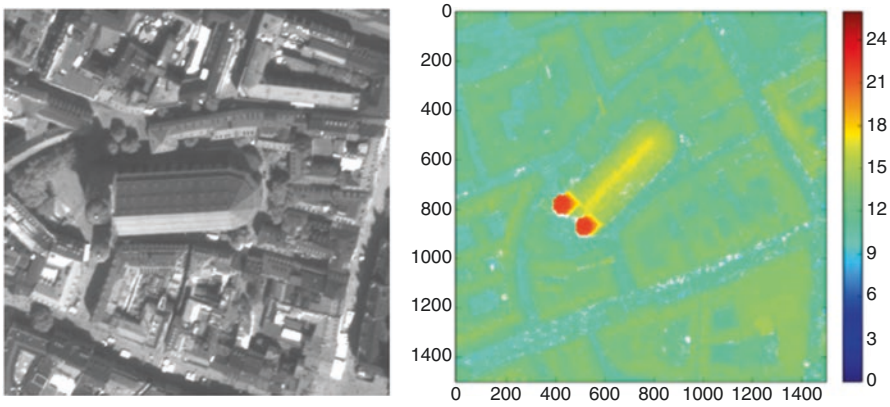

Fig. 15 Original satellite image (left) and the finally derived georeferenced DSM (right, courtesy Pablo d'Angelo)

The geographic referenced DSM is generated directly in the same step or later from this height map M by calculating λ and φ for each point (x, y) of the original image I_1^{orig} using the corresponding height $h = M(x, y)$ by iteratively inverting the sensor model.

Examples

Munich, Historic Center

The first example covers the historic city center from Munich, Germany. The stereo images were acquired 10 July 2012, 10:30 UTC by the satellite WorldView-2 (©2012 European Space Imaging). The whole images cover an area of about 20 × 20 square kilometers. In Fig. 16 the selected area of the historic city center is shown.

A WorldView-2 image consists always of a multispectral image with a ground resolution of about 2 m containing eight spectral bands (coastal, blue, green, yellow, red, red-edge, and two near-infrared bands) and one panchromatic image of 0.5 m resolution and one band. The pan bands of the two stereo images are usually used for the generation of the DSM. The satellite images are accompanied by the RPCs representing the sensor model. In the first step, the area of interest has to be cut from both of the stereo images and transformed to epipolar geometry as shown in Fig. 17.

Afterward the disparity map is calculated on the epipolar images using the SGM method and transformed back as height map to the original image. After filling occlusions and mismatches, the final DSM can be seen in Fig. 18, right.

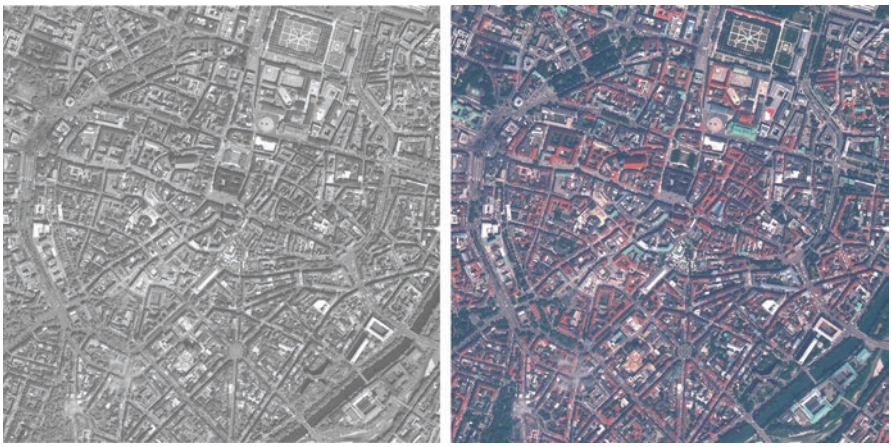

Fig. 16 Section 1.75 × 1.75 km^2, Munich city center, left, panchromatic "left" image; right, multispectral (red, green, blue) "left" image

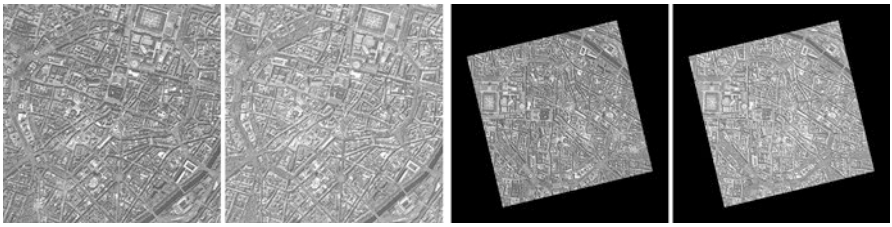

Fig. 17 Section 1.75 × 1.75 km^2, Munich city center, left and right original panchromatic images, left and right epipolar images used for DSM generation

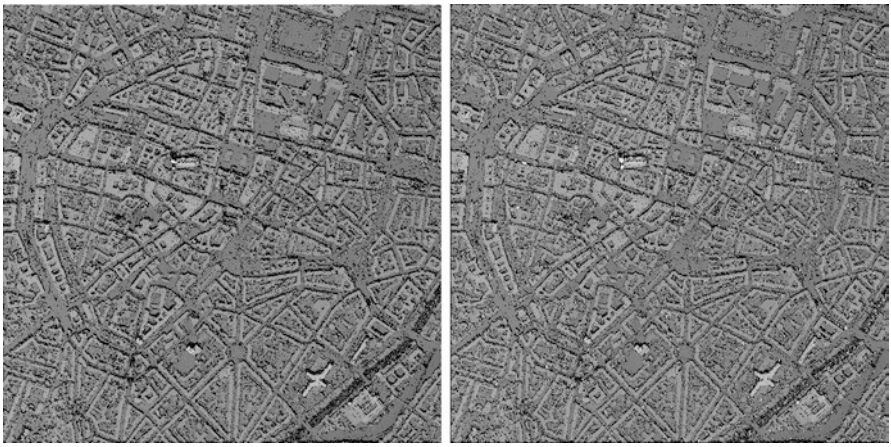

Fig. 18 Section 1.75 × 1.75 km^2, Munich city center, left, height map fitting on left original image; right, final DSM, UTM zone 32 north, WGS84 ellipsoid

Vaihingen, Historic Center

In the scope of a EuroSDR benchmark for generating DSMs and urban models from aerial imagery, stereo images were captured by a UltraCam-X at a height above ground of 2900 m and a ground sampling distance (GSD) of 20 cm over Vaihingen/Enz in Germany. The DSM is generated from the oblique view images as described above using the semi-global matching method. Figure 19 shows the two panchromatic stereo images together with the derived SGM height map fitting on the first of the images.

Athens, Acropolis

Another example for a DSM derived from satellite imagery is the area around the Acropolis in Athens as shown in Fig. 20. The stereo image was acquired by the former Ikonos-2 satellite at 24th of July 2004, 9:25 UTC with a ground resolution of about 1 m. The image was shipped already in epipolar format together with RPCs as sensor model.

Fig. 19 Section 400 × 400 m^2, Vaihingen city center, left/center, left and right stereo image; right, derived height map

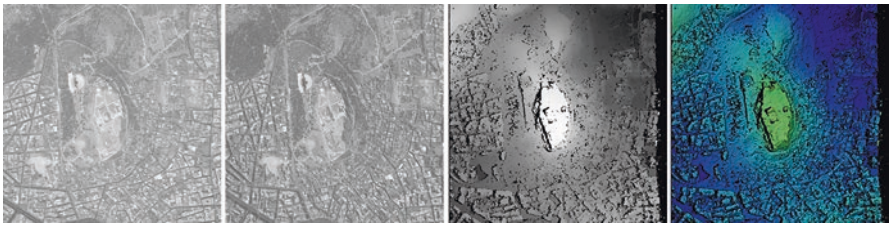

Fig. 20 Section 1 × 1 km^2, Athens, Acropolis, satellite image Ikonos, 24 July 2004, left and right stereo image, height map and color-coded and shaded height map

Cairo, Pyramids

The last example is a stereo pair taken by the satellite GeoEye at 2nd of July 2009, 8:47 UTC over Cairo, Egypt. The ground resolution of the panchromatic sensor of this satellite is also about 0.5 m. Figure 21 shows a smaller section around the pyramids from the 12 × 12 km² satellite scene.

From the section of Fig. 21, a small area around the pyramids was selected to calculate height maps using the SGM algorithm. Figure 22 shows the derived epipolar images.

Due to the large off-nadir angles of 14° and 21° of the stereo images, two SGM height maps were calculated to illustrate the differences, one fitting on the left epipolar image and the other fitting on the right epipolar image. The resulting height maps are shown in Fig. 23 and their color-coded and shaded counterparts in Fig. 24.

As can be seen in Fig. 23, left, the right (southern) surface of the Cheops pyramid has nearly the same inclination as the viewing angle of the south viewing stereo image of the satellite. In such cases where areas can only be nicely seen in one of the two stereo images, no stereo matching is possible, and the area is marked as "no data." Such areas were also visible in all of the other (unfilled) height maps or dense DSMs presented in this chapter. But mostly these areas occur near steep walls in cities and can easily be filled using the nearby ground heights.

Fig. 21 Section 2.5 × 2.5 km², Cairo, GeoEye, 2 July 2009, left and right stereo image

Fig. 22 Section 1 × 0.5 km², Cairo, GeoEye, 2 July 2009, left and right stereo image

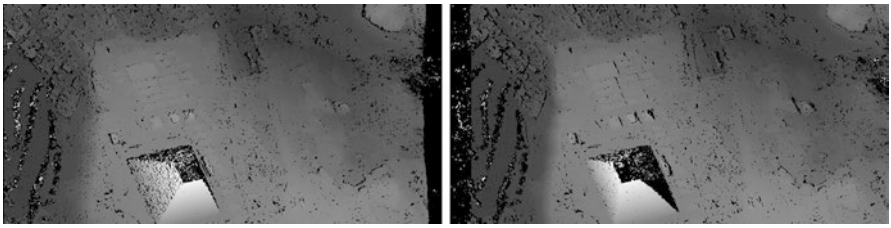

Fig. 23 Section 1 × 0.5 km², Cairo, height maps fitting on first (left) and second (right) epipolar stereo image

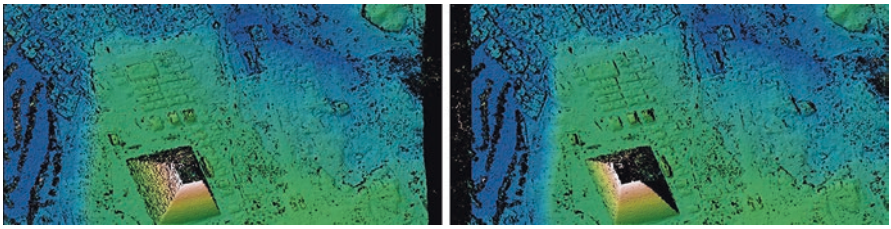

Fig. 24 Section 1 × 0.5 km², Cairo, height maps fitting on first (left) and second (right) epipolar stereo image, shaded and heights color-coded

Conclusions

In this chapter we showed how dense digital surface models (DSMs) could be generated from (multi-)stereo satellite or airborne imagery. Starting with the basics, the key methods up to the actually best semi-global matching (SGM) were described and the whole chain of DSM generation explained. The presented methods can be applied to a wide range of stereo images. The only prerequisite is having a valid sensor model of the camera used. So starting from stereo images taken with hand-held or fixed cameras above an excavation pit over small drones covering only a small part of an archeological site to planes or even satellites mapping 16 × 16 square kilometers at once, high-quality surface models in all scales of resolution and coverage can be generated in reasonable good quality, mostly in the scale of the ground resolution of the camera used.

References

Birchfield S, Tomasi C (1998) Depth discontinuities by pixel-to-pixel stereo. In: Proceedings of the 1998 IEEE international conference on computer vision, pp 1073–1080

d'Angelo P, Lehner M, Krauß T, Hoja D, Reinartz P (2008) Towards automated DEM generation from high resolution stereo satellite images. In: IAPRS, vol 37, pp 1137–1142

Grodecki J, Dial G, Lutes J (2004) Mathematical model for 3D feature ex-traction from multiple satellite images described by RPCs. In: ASPRS annual conference proceedings, Denver

Hirschmüller H (2005) Accurate and efficient stereo processing by semi-global matching and mutual information. In: IEEE conference on computer vision and pattern recognition (CVPR), vol 2, pp 807–814

Jacobsen K, Byksalih G, Topan H (2005) Geometric models for the orientation of high resolution optical satellite sensors. In: ISPRS workshop, Hannover, vol 36 (1/W3)

Krauß T, Reinartz P, Lehner M, Schroeder M, Stilla U (2005) DEM generation from very high resolution stereo satellite data in urban areas using dynamic programming. In: ISPRS workshop, Hannover, vol 36 (1/W3)

Lehner M, Gill RS (1992) Semi-automatic derivation of digital elevation models from stereoscopic 3-line scanner data. Int Arch Photogramm Remote Sens 29(B4):68–75

Otto GP, Chau TKW (1989) Region growing algorithm for matching of terrain images. Image Vis Comput 2(7):83–94

Scharstein D, Szeliski R (2002) A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int J Comput Vis (IJCV) 47(1/2/3):7–42

Spangenberg R, Langner T, Rojas R (2013) Weighted semi-global matching and center- symmetric census transform for robust driver assistance. Comput Anal Images and Patterns 8048:34–41

van Meerbergen G, Vergauwen M, Pollefeys M, van Gool L (2002) A hierarchical symmetric stereo algorithm using dynamic programming. Int J Comput Vis 47(1/2/3):275–285

Zabih R, Woodfill J (1994) Non-parametric local transforms for computing visual correspondence. In: Computer vision – ECCV, vol 1994, pp 151–158