Orthophoto
Orthophoto

• **Definition**
  • an orthophoto is an image of the object surface in **orthogonal parallel Projection**
  • can be the surface of the earth, but also a building facade, a technical component ...
  • Orthophotos are usually generated synthetically (actually recording an orthogonal parallel projection is hard)

• **Ingredients**
  • an image of the surface of interest
  • interior and exterior orientation of the image
  • a digital model of the surface - for the earth, a digital surface model (DSM)
  • **Note**: here only digital images are treated; before the advent of digital image processing there were analog machines for ortho-rectification
Orthophoto

• Example
Rectification

- Starting from the original image
  - project image to surface model
  - project textured surface model onto \((X,Y)\)-plane

- Starting from orthophoto plane
  - for each point \(x_0=(X,Y)\) of the orthophoto
    - determine the surface point \(X=(X,Y,Z(x))\)
    - map the surface point to the original camera image \(x=PX\)
    - read out the brightness values for the orthophoto: \(I(x_0) = I(x)\)
Rectification

• in a discrete pixel raster
  • Define pixel positions in orthophoto
  • interpolate surface height at those positions
  • project 3D surface points to image points
  • interpolate brightness values for the orthophoto from the neighborhood in the original image
DTM Interpolation

- Interpolation of surface (terrain) heights
  - at every position in the pixel raster of the orthophoto one needs to determine a height

- Example
  - Surface model in raster form (coarser than orthophoto)
  - bilinear interpolation

\[
z = (1 - \Delta x) \cdot (1 - \Delta y) \cdot z_{00} + \Delta x \cdot (1 - \Delta y) \cdot z_{01} + (1 - \Delta x) \cdot \Delta y \cdot z_{10} + \Delta x \cdot \Delta y \cdot z_{11}
\]
Resampling

• Estimation of brightness values $I_{ortho}$ for the orthophoto
  • for each raster point $X_{ortho}$ on the surface
    - project to camera image with collinearity $x_{cam} = P X_{ortho}$
    - compute intensity $I(x_{cam})$ from surrounding intensity values
    - invert projection, i.e. assign $I(x_{ortho}) = I(x_{cam})$
  
• in multi-channel images: separately get intensity for each channel
Resampling

• Estimation of brightness values $I_{ortho}$ for the orthophoto
  • for each raster point $X_{ortho}$ on the surface
    - project to camera image with collinearity $x_{cam} = P \cdot X_{ortho}$
    - compute intensity $I(x_{cam})$ from surrounding intensity values
    - invert projection, i.e. assign $I(x_{ortho}) = I(x_{cam})$

• in multi-channel images: separately get intensity for each channel
Resampling

- Estimation of brightness values $I_{ortho}$ for the orthophoto
  - for each raster point $X_{ortho}$ on the surface
    - project to camera image with collinearity $x_{cam} = P \cdot X_{ortho}$
    - compute intensity $I(x_{cam})$ from surrounding intensity values
    - invert projection, i.e. assign $I(x_{ortho}) = I(x_{cam})$

- in multi-channel images: separately get intensity for each channel
Resampling

- Estimation of brightness values $I_{ortho}$ for the orthophoto
  - for each raster point $X_{ortho}$ on the surface
    - project to camera image with collinearity $x_{cam} = P \cdot X_{ortho}$
    - compute intensity $I(x_{cam})$ from surrounding intensity values
    - invert projection, i.e. assign $I(x_{ortho}) = I(x_{cam})$

- in multi-channel images: separately get intensity for each channel
Resampling

• Estimation of brightness values $I_{ortho}$ for the orthophoto
  • for each raster point $X_{ortho}$ on the surface
    - project to camera image with collinearity $x_{cam} = P \cdot X_{ortho}$
    - compute intensity $I(x_{cam})$ from surrounding intensity values
    - invert projection, i.e. assign $I(x_{ortho}) = I(x_{cam})$

• in multi-channel images: separately get intensity for each channel
Resampling

- Estimation of brightness values $I_{ortho}$ for the orthophoto
  - for each raster point $X_{ortho}$ on the surface
    - project to camera image with collinearity $x_{cam} = P X_{ortho}$
    - compute intensity $I(x_{cam})$ from surrounding intensity values
    - invert projection, i.e. assign $I(x_{ortho}) = I(x_{cam})$

- in multi-channel images: separately get intensity for each channel
Resampling

- Intensity values off the pixel raster
  - nearest neighbor: value of nearest known point
  - original intensities are preserved
  - image structures are shifted → aliasing!

\[
I(x, y) = \begin{cases} 
I(x_-, y_-) & \text{if } (\Delta x \leq 0.5) \land (\Delta y \leq 0.5) \\
I(x_-, y_+) & \text{if } (\Delta x \leq 0.5) \land (\Delta y > 0.5) \\
I(x_+, y_-) & \text{if } (\Delta x > 0.5) \land (\Delta y \leq 0.5) \\
I(x_+, y_+) & \text{if } (\Delta x > 0.5) \land (\Delta y > 0.5) 
\end{cases}
\]
Resampling

- Intensity values off the pixel raster
  - nearest neighbor: value of nearest known point
  - original intensities are preserved
  - image structures are shifted → aliasing!

\[
I(x, y) = \begin{cases} 
(\Delta x \leq 0.5) \land (\Delta y \leq 0.5) : & I(x-, y-) \\
(\Delta x \leq 0.5) \land (\Delta y > 0.5) : & I(x-, y+) \\
(\Delta x > 0.5) \land (\Delta y \leq 0.5) : & I(x+, y-) \\
(\Delta x > 0.5) \land (\Delta y > 0.5) : & I(x+, y+) 
\end{cases}
\]
Resampling

- Intensity values off the pixel raster
  - bilinear interpolation in a (2 x 2) neighborhood
  - linear in each coordinate, but interpolant is 2nd order surface
  - noticeable smoothing

\[
I(x, y) = (1 - \Delta x) \cdot (1 - \Delta y) \cdot I(x_-, y_-) + \\
\Delta x \cdot (1 - \Delta y) \cdot I(x_+, y_-) + \\
(1 - \Delta x) \cdot \Delta y \cdot I(x_-, y_+) + \\
\Delta x \cdot \Delta y \cdot I(x_+, y_+)
\]
Resampling

- Intensity values off the pixel raster
  - bilinear interpolation in a (2 x 2) neighborhood
  - linear in each coordinate, but interpolant is 2nd order surface
  - noticeable smoothing

\[
I(x, y) = (1 - \Delta x) \cdot (1 - \Delta y) \cdot I(x-, y-) + \\
\Delta x \cdot (1 - \Delta y) \cdot I(x+, y-) + \\
(1 - \Delta x) \cdot \Delta y \cdot I(x-, y+) + \\
\Delta x \cdot \Delta y \cdot I(x+, y+)
\]
Resampling

- Intensity values off the pixel raster
  - bicubic interpolation in a (4 x 4) neighborhood
  - cubic polynomial in each coordinate
  - theoretically still not a smooth interpolant, but in practice very good

\[
I(x, y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} \cdot (1 + \Delta x)^i \cdot (1 + \Delta y)^j
\]

\[
a_{ij} = f(I(x-, y-), I(x-, y-), \ldots, I(x++, y++))
\]
Resampling

- Intensity values off the pixel raster
  - bicubic interpolation in a (4 x 4) neighborhood
  - cubic polynomial in each coordinate
  - theoretically still not a smooth interpolant, but in practice very good

\[
I(x, y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} \cdot (1 + \Delta x)^i \cdot (1 + \Delta y)^j
\]

\[
a_{ij} = f(I(x-, y-), I(x-, y-), \ldots, I(x++, y++))
\]
Planar Rectification

- planar object surface
  - mapping from orthophoto plane to object plane is a parallel projection, thus affine
    6 parameters: position, orientation, anisotropic scaling and shear of coordinate axis
  - mapping between object plane and image plane is line-preserving and 2D-to-2D, so a projectivity (pinhole camera)
    8 parameters for homography
  - combination is also a projectivity

\[
x' = AH_s x = Hx
\]

\[
A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\
               a_{21} & a_{22} & a_{23} \\
               0       & 0       & a_{33} \end{bmatrix}
\]

\[
H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\
               h_{21} & h_{22} & h_{23} \\
               h_{31} & h_{32} & h_{33} \end{bmatrix}
\]
Planar Rectification

- Homography from orientation parameters
  
  assumption: object in \((X,Y)\)-plane (can always be reached by a Helmert transformation)

\[
x = PT \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \tilde{P} \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \tilde{p}_{11} & \tilde{p}_{12} & \tilde{p}_{14} \\ \tilde{p}_{21} & \tilde{p}_{22} & \tilde{p}_{24} \\ \tilde{p}_{31} & \tilde{p}_{32} & \tilde{p}_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}
\]

\[
H = \begin{bmatrix} \tilde{p}_{11} & \tilde{p}_{12} & \tilde{p}_{14} \\ \tilde{p}_{21} & \tilde{p}_{22} & \tilde{p}_{24} \\ \tilde{p}_{31} & \tilde{p}_{32} & \tilde{p}_{34} \end{bmatrix}^{-1}
\]

\[
x' = Hx
\]

- **Note**: other than perspective projection, the projectivity is one-to-one (bijective) and thus invertible
Planar Rectification

- direct estimation of the homography
  - all we need are the 8 unknowns of $H$
  - a pair of corresponding points yield 2 linearly independent equations
  - $\geq 4$ correspondences (no 3 of them on a line) yield $\geq 8$ equations, solution via singular value decomposition
  - neither the interior nor the exterior orientation need to be known!

$$\lambda x' = Hx \quad Hx \times x' = 0$$

$$\begin{bmatrix} 0 & 0 & 0 & x & y & 1 & -xy' & -yy' & -y' \\ -x & -y & -1 & 0 & 0 & 0 & xx' & yx' & x' \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$
Planar Rectification

- Example: facade orthophoto
Orthophoto - Accuracy

• Two main sources of error

• inaccurate DSM
  • height errors in the surface model are mapped to planimetric errors in the orthophoto
  • points are displaced in radial direction w.r.t. the nadir

• Occlusions
  • string relief of the surface leads to occlusions
  • the intensities required for the orthophoto are not found in the original image
Orthophoto - Accuracy

- inaccurate DSM
  - often happens when of-the-shelf terrain models (DTMs) are used, especially if they do not contain buildings and vegetation
  - nowadays less frequent because of the increased availability of high-quality surface models (LiDAR, dense image matching) and 3D city models
Orthophoto - Accuracy

• Radial displacement
  • Vertical errors in the surface model are mapped to radial planimetric errors
  • the error depends on
    - the magnitude of the height error
    - the orientation
    - the point’s position in the image
    - the surface slope
  • for approximate nadir views and flat terrain

\[ R = H \cdot \tan \beta = \frac{r}{c} \cdot H \]
Orthophoto - Accuracy

- Radial displacement increases with
  - increasing distance from the principal point / nadir (i.e. in image blocks also with decreasing overlap → border areas of images have to be used)
  - smaller camera constant resp. lower flying height

\[ R = \frac{r}{c} \cdot H \]
Orthophoto - Accuracy

• Occlusions
  • strong relief leads to occlusions
  • brightness values for regions that are visible in the orthophoto (“from above”) are not observable in the original image
  • in weaker form: area is visible, but perspective foreshortening is too strong to extract usable texture (especially near the border of wide-angle images)
True Orthophoto

• Solution
  • Detection of occluded areas using the DSM
  • Use of multiple images to obtain an orthophoto without holes
  • Often called “true orthophoto”