3D RECOVERING OF URBAN SCENES

Juan Felipe Montesinos, Yi Xiao, Ferran Carrasquer

2D homography

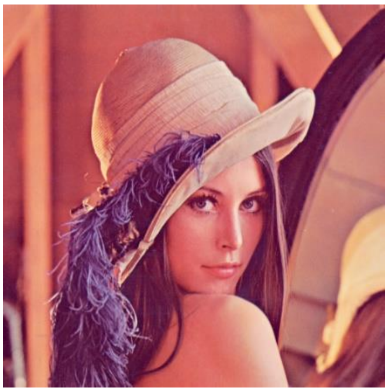
A 2D homography can be computed in $\mathbb{P}^2$ space through an homogeneus transform following general expression:

$$x' = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{12} & h_{22} & h_{23} \\ h_{13} & h_{23} & 1 \end{pmatrix} x$$

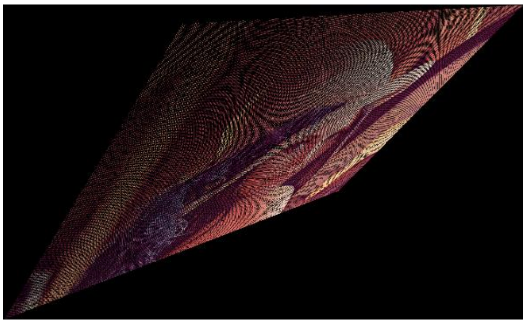
$$x' = \begin{pmatrix} A & t \\ v & 1 \end{pmatrix} x$$

Applying an homography to an image is equivalent to applying the homography to each pixel.

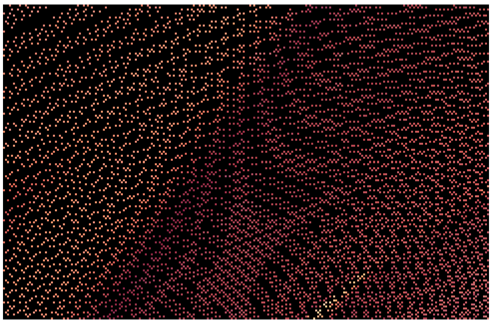
Since images are defined in $\mathbb{N}^2$ there are not infinite points avilable to carry out a perfect homography. Artifacts appears due to quantization effect of rounding real numbers into natural numbers as show in the figure 1.



Original image



Transformed image



Quantization effect

Depending on the final size of the image this transformation can be injective or suprajective depending on the final size and deformation. For solving the quantization problem an inverse mapping is applied. This way, since homograpies are non-singular matrices it is possible to assign a value to each pixel mapping according the suprajective transformation.

Computational considerations

Matrices are computally talking multidimensional arrays of integer numbers (usually) where array indexation (which are mandatory natural numbers) represent the position of each pixel in the image. Homographies may transform points into negative positions. Therefore it is necessary to apply a translation from integer space to natural space in order to be computable.

Given an image of size M x N the translation $\vec{t}$ to go from integer-numbers space to natural-numbers space is:

$$\min_{x,y}(\vec{a}, \vec{b}, \vec{c}, \vec{d})$$

$$\vec{a} = H(1\ 1\ 1)'$$

$$\vec{b} = H(1\ N\ 1)'$$

$$\vec{c} = H(M\ 1\ 1)'$$

$$\vec{d} = H(M\ N\ 1)'$$

3D RECOVERING OF URBAN SCENES

Juan Felipe Montesinos, Yi Xiao, Ferran Carrasquer

Result and Comments

Applying image transformations

Function *V2H(inputs)* has been created generate homographies given basic parameters such as projection vector, v; affinity submatrix, A; translation vector, t as inputs. Whatever input order is allowed. If some parameter is not set as input function set it by default by its invarian form.

Similarity

Similarities consist of a rotation and isotropic scaling. 2D rotation matrix in Euclidean space are computed by the function angle2R, considering counter clockwise rotation. For camparision, two examples with the angles of 20 and 40 degrees are showed as below:

Similarity (angle=20)

Similarity (angle=40)

Transformed-image sizes are automatically adjusted.

Affine transformation

Affinities consists of a rotation and a anisotropic scaling perfomed in an specific orientation.

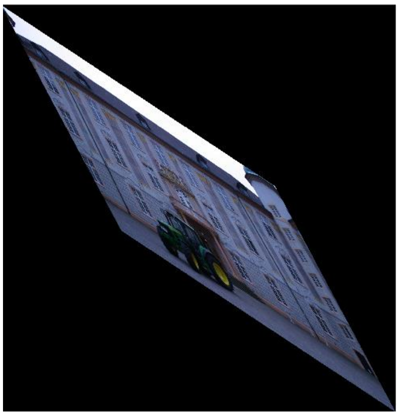
$$R_\theta R_{-\phi} D R_\phi$$

The first angle Θ is for the rotation of the whole image. The second angle is for scaling the image in an specific orientation. The scaling effect depends on D, which is a 2x2 diagonal matrix.

Affine transformation (Θ=0 and ∅=40, D=[2 1])

Affine transformation (Θ=0 and ∅=40, D=[1 2])

Notice that negative scaling parameters imply mirroring .



Affine transformation (ϴ=0 and ∅=40, D=[1 4])

Affine transformation (ϴ=40 and ∅=40, D=[1 1])

D determines the scaling effect. If the D is set as [1 1], the second angle ∅ has no influence on the image, finally performing a rotation.

Projective transformation

3x3 matrices have 9 degrees of freedom. Since projective transformations are carried out in homogenus coordinates only 8 dregrees of freedom are needed to define an homography. Therefore any invertible 3 by 3 matrix defines a projective transformation of aplane.

Projective transformation

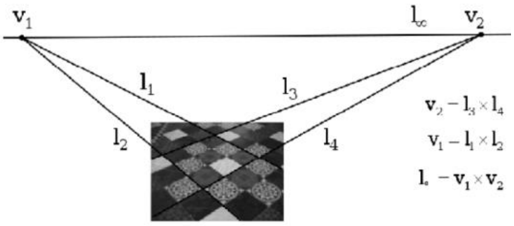
3D RECOVERING OF URBAN SCENES

Juan Felipe Montesinos, Yi Xiao, Ferran Carrasquer

2-step image rectification

Affine Rectification (Paralell-line-based)

Affine transformation is a tecnique for converting a projection into an affinity- It is computed by finding the vanishing line at the infinity defined by two vanishing points, which are intersection of two pairs of parallel images.



The homography which brings the infinity line back to the infinity (this means, converting the projection into an affinity) is the following one, being Ha any affine homography.

$$\mathbf{H}_{\mathbf{a}\leftarrow\mathbf{p}} = H_a \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{pmatrix}$$

Original image with the pair of parallel lines selected to extract the infinite line:

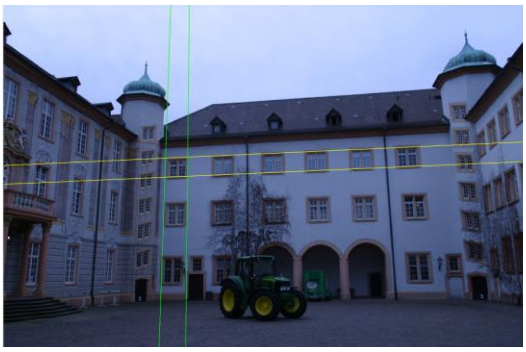
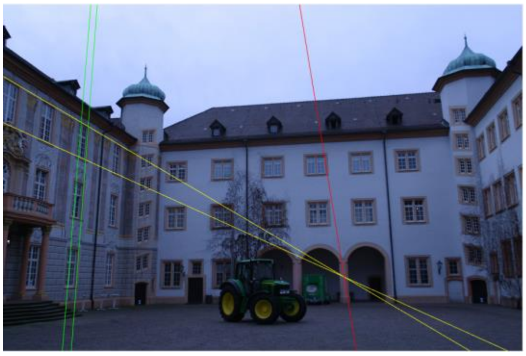
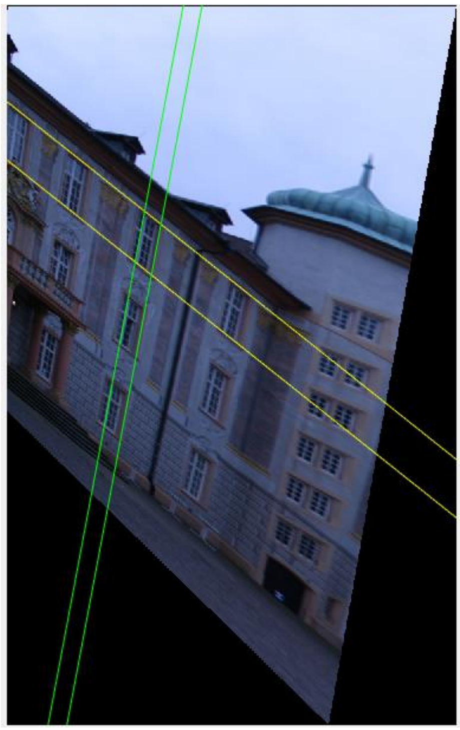


Image affine rectified:

Since the line at infinity and vanishing points are at a non-ideal point in a projection, it is not possible to compute points beyond the infinity line (red line).



Unprojecting left facade:



3D RECOVERING OF URBAN SCENES

Juan Felipe Montesinos, Yi Xiao, Ferran Carrasquer

Metric Rectification

Once affine rectification has been carried out, metric rectification can be applied.

Metric rectification can be line-based performanced using 2 pairs of non-parallel orthogonal lines.

General expression to transform an affinity into a similarity is the following one:

$$H_{a\leftarrow s} = \begin{pmatrix} K & \vec{0} \\ \vec{0}^T & 1 \end{pmatrix}$$

Since the images of two orthogonal lines in the real world satifies the following equation:

$$(l_1 m_1, l_1 m_2 + l_2 m_1, l_2 m_2)\vec{s} = 0.$$

It is necessary two equations to solve the system, where s=(s1 s2 s3) is the nullspace of the product

$$S = AA^T = \begin{pmatrix} s_1 & s_2 \\ s_2 & s_3 \end{pmatrix}.$$

Where K=A can be reached by using Cholesky decomposition

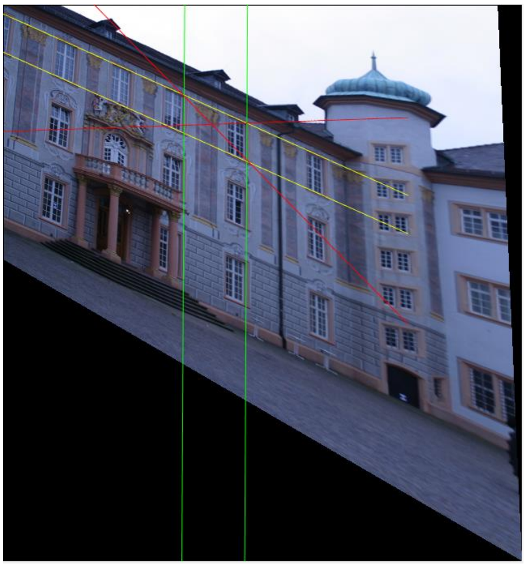
By finally get the H matrix that will use in order to do the metric rectification:

$$H_{\mathrm{a}\leftarrow \mathrm{s}}^{-1} = H_{\mathrm{s}\leftarrow \mathrm{a}} = \begin{pmatrix} K^{-1} & \vec{0} \\ \vec{0}^T & 1 \end{pmatrix}$$

It is important to highlight that if both couples or orthogonal lines are parallel, it is not possible to compute the homography since a non-independent system of equations is reached.

Results of the purposed rectification to the left facade:

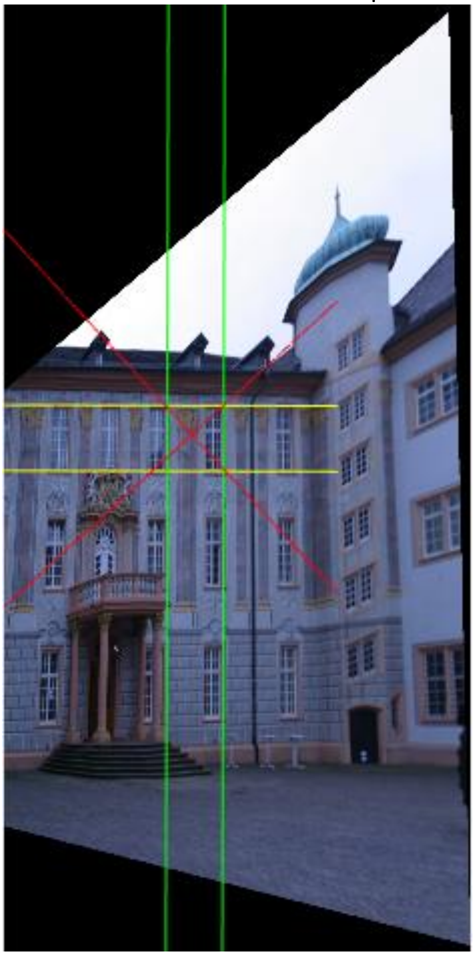
Original image and lines used



Affine-rectified image

3D RECOVERING OF URBAN SCENES

Juan Felipe Montesinos, Yi Xiao, Ferran Carrasquer



Rectified image

1-step image rectification

The approach for carrying out the 1-step rectification is similar to the metric rectification. 5 pairs of orthogonal lines are required in order to solve a linear system of equations based in the same concept as metric rectification, which brings to the following expression:

$$\begin{pmatrix} KK^T & KK^T\vec{v} \\ \vec{v}^T KK^T & \vec{v}^T KK^T \vec{v} \end{pmatrix}$$

Where K and v are the affinity submatrix and vanishing line vector respectively.

However for the purposed image there are not 5 pair of non-parallel orthogonal lines.

The output obtained from LSD detector is the following one:

In which it is not possible to (easy) appreciate lines under required constriction..