BMVC2000

A new Approach for Vanishing Point Detection in Architectural Environments

Carsten Rother
Computational Vision and Active Perception Laboratory (CVAP)
Department of Numerical Analysis and Computer Science
Royal Institute of Technology (KTH), S–10044 Stockholm, Sweden
carstenr@nada.kth.se

Abstract

A man-made environment is characterized by a lot of parallel lines and a lot of orthogonal edges. In this article, a new method for detecting the three mutual orthogonal directions of such an environment is presented. Since real-time performance is not necessary for architectural application, like building reconstruction, a computationally more intensive approach was chosen. On the other hand, our approach is more rigorous than existing techniques, since the information given by the condition of three mutual orthogonal directions in the scene is identified and incorporated. Since knowledge about the camera geometry can be deduced from the vanishing points of three mutual orthogonal directions, we use this knowledge to reject falsely detected vanishing points. Results are presented from interpreting outdoor scenes of buildings.

Key words

Vanishing points, vanishing lines, geometric constraints, architecture, camera calibration

1 Introduction

The analysis of vanishing points provides strong cues for inferring information about the 3D structure of a scene. With the assumption of perfect projection, e.g. with a pin-hole camera, a set of parallel lines in the scene is projected onto a set of lines in the image that meet in a common point. This point of intersection, perhaps at infinity, is called the *vanishing point*. Vanishing points, which lie on the same plane in the scene, define a line in the image, so-called the *vanishing line*. Figure 1 shows the three vanishing points and vanishing lines of a cube, where a finite vanishing point is defined by a point on the image plane and a vanishing point at infinity is defined by a direction on the image plane. When the camera geometry is known, each vanishing point corresponds to an orientation in the scene and vice versa.

The understanding and interpretation of man-made environments can be greatly simplified by the detection of vanishing points. This has been done e.g. in the field of navigation of robots and autonomous vehicles [12], in the field of object reconstruction [5] or

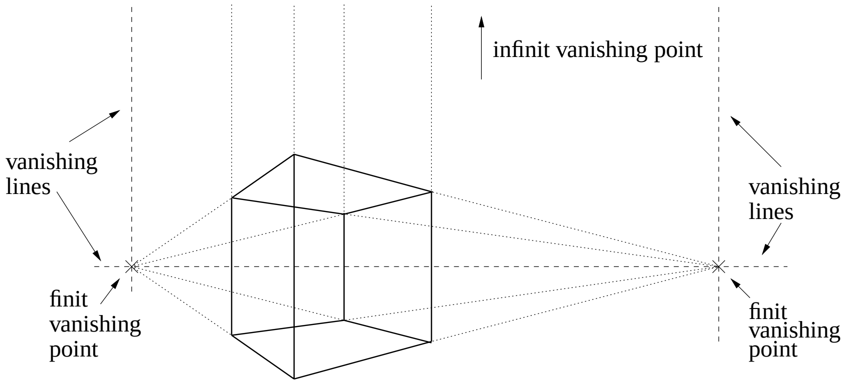


Figure 1: The three vanishing points and vanishing lines of a cube.

for the calibration of cameras [4, 8, 13]. A man-made environment has two characteristic properties: A lot of lines in the scene are parallel and a lot of edges in the scene are orthogonal. In an indoor environment this is true for e.g. shelfs, doors, windows and corridor boundaries. In an outdoor environment e.g. streets, buildings and pavements satisfy this assumption. On the basis of these properties the task of detecting the three mutual orthogonal directions of a man-made environment has raised a lot of interest [6, 7, 12].

After a discussion of existing vanishing point detection methods in section 2, our method is presented in section 3. Section 4 demonstrates the performance of our method on real image data.

2 Previous Work

The majority of vanishing point detection methods relies on line segments detected in the image. A different approach is to consider the intensity gradients of the pixel units in the image directly [6, 14]. Since we base our method on line segments, these approaches will be considered in the following in more detail.

The task of detecting those vanishing points, which correspond to the dominant directions of a scene, is traditionally solved in two steps. Firstly, line segments are clustered together under the condition that a cluster of line segments share a common vanishing point. We denote this step as *accumulator step*. In the second step the dominant clusters of line segments are searched for. We refer to this step as the *search step*.

Let us consider the accumulation step first. In order to reduce the computational complexity of the clustering process, the unbounded image R^2 is mapped onto a bounded space. This has the additional advantage that infinite and finite vanishing points can be treated in the same way. The bounded space, also denoted as *accumulator space*, can then be partitioned into a finite number of cells, so-called *accumulator cells*.

Barnard [2] suggested the Gaussian sphere centred onto the optical centre of the camera as an accumulator space (see figure 2). A great circle on the Gaussian sphere represents a line segment in the image and a point on the Gaussian sphere corresponds to a vanishing point in the image. Figure 2 shows that the great circles of two line segments in

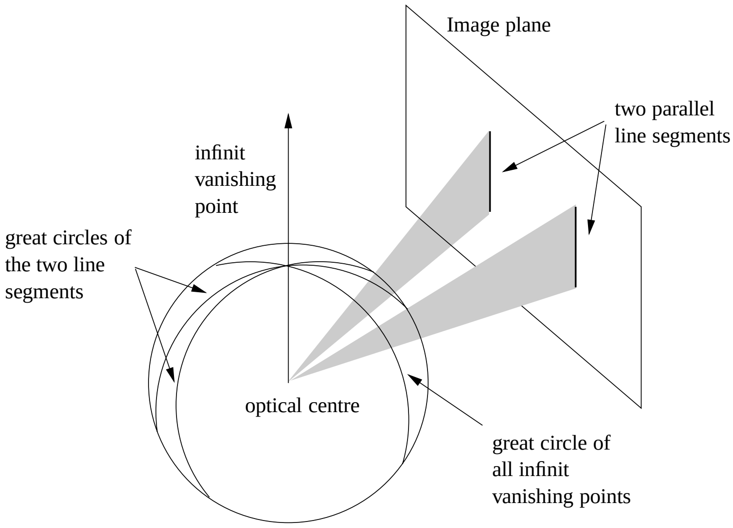


Figure 2: The Gaussian sphere as an accumulator space.

the image plane intersect always in one point, their vanishing point. For the accumulation of line segments, the Gaussian sphere is tessellated into accumulator cells, and each cell is increased by the number of great circles which pass through it.

This approach was then enhanced in other works. Since Barnard chose an irregular and quite ad hoc tessellation of the Gaussian sphere, this was improved by Quan and Mohr [11]. Lutton [9] investigated the influence of different error types, e.g. error due to the finite extension of the image, in the accumulation process on the Gaussian sphere. Magee and Aggarwal [10] accumulated the projection of the intersection points of all pairs of line segments in the image onto the Gaussian sphere. This approach is computationally more intensive but on the other hand more accurate.

Alternative accumulator spaces were introduced by the authors [12, 3]. Brillault [3] established an uncertainty model for a line segment. According to this model an accumulator space is introduced, in which the expected uncertainty of a line segment remains constant over the accumulator space.

A different approach for reducing the computational complexity of the accumulation step is to apply the Hough transformation by mapping the parameters of the line segments into a bounded Hough space [1, 14]. Tuytelaars et al. [14] applied the Hough transformation three times (Cascade Hough transformation). At different levels of the Cascade Hough transformation a peak in the Hough space corresponds to a vanishing point and a vanishing line respectively.

A main drawback of all techniques which transfer information from the image into a bounded space is that the original distances between lines and points are not preserved. Let us consider the two great circles of the two line segments in figure 2. Due to the perspective effect of the projection from the image plane onto the Gaussian sphere the distance between these two great circles differs when the two line segments undergo the same movement on the image plane. Therefore, the distance between a line segment and a vanishing point depends on their location on the image plane, i.e. the distances between points and lines on the image plane are not translationally and rotationally invariant. This drawback can be avoided when line segments are not transformed into a bounded space, i.e. the image plane is chosen as the accumulator space.

In the past, more effort has been spent on the accumulation step than on the search step. One of the reasons for this is that the directions in the scene of the searched dominant vanishing points did not have to be orthogonal. This means that the orthogonality of the direction of vanishing points was not treated as an additional criterion of the search step. In [10, 11] the search step was designed in a straight forward manner. Firstly, the dominant vanishing point, which corresponds to the accumulator cell with the most line segments, is detected. After removing the line segments, which correspond to this vanishing point, the search for a maximum in the accumulator space is repeated. This iterating process stops when the number of line segments of a dominant vanishing point is below a certain threshold. This approach is characterized by a minimal computational effort.

Recently van den Heuvel [7] developed a method for detecting the three mutual orthogonal directions in the scene. The orthogonal criterion was explicitly used, which means that all combinations of three possible vanishing points have to be considered. This higher computational effort is the main drawback compared to the approach mentioned above. However, van den Heuvel chose one of the vanishing points manually. Coughlan and Yuille [6] searched for two orthogonal directions in the scene by using statistics.

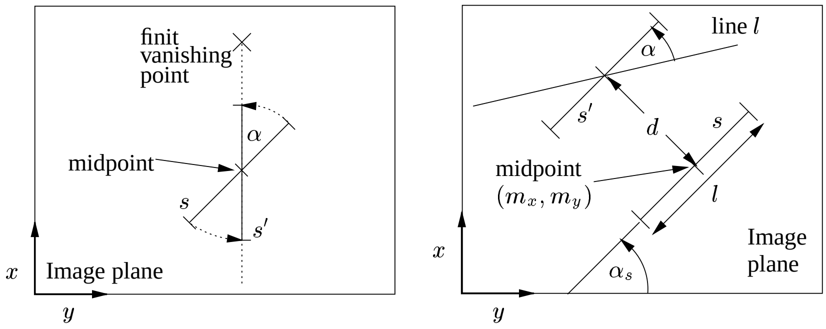
3 Detection of three mutual orthogonal directions

With increased computing power and without the condition of real-time performance an approach with higher computational effort is reasonable and will be pursued here. The basic idea of our approach is to establish a coherent and simple framework (like [7]) which can be used for the accumulation step as well as for the search step. As accumulator space we choose the unbounded image plane itself, which has the advantages as we already pointed out above. As accumulator cells we choose (like [10]) the intersection points of all pairs of line segments. We will see that despite the fact that the image plane is unbounded, infinite and finite vanishing points can be treated in the same way.

For the search step we will establish a number of criteria, which vanishing points with mutual, orthogonal directions have to satisfy. Since one of the criteria is the orthogonal criterion, we obtain a higher computation effort than an easier approach (like [10, 11]).

3.1 Accumulation Step

Due to various reasons is the perspective projection of a line segment from the 3D scene onto the 2D image not congruent with the line segment detected in the image. We denote this perfect projection of a line segment as *projected line segment*. Hence, all vanishing points detection methods have to formulate either implicitly or explicitly a distance function between a vanishing point and a detected line segment. In this context the basic question is: *How close is a projected line segment s' with vanishing point vp to its corresponding line segment s.* In order to answer the question we represent a line segment with the midpoint representation (m_x, m_y, l, α_s) (see figure 3 (b)). Compared to other representations, e.g. endpoint representation, it has the advantage that the length of a line segment is treated explicitly. We define: The perfect line segment s' of a line segment s



(a) (b)

Figure 3: Explanation for the distance function $d(vp, s)$ between a line segment s and a finite vanishing point vp (a), the midpoint representation of a line segment s (b) and the distance function $d(l, s)$ between a line l and a line segment s (b).

has the same midpoint as s and has vp as vanishing point. On the basis of this definition a distance function $d(vp, s)$ between a vanishing point vp and a line segment s can be defined as the angle α between the corresponding line segments s' and s. Figure 3 (a) gives an example for a finite vanishing point. Since we need a distance function $d(l, s)$ between a line l and a line segment s in the search step, we define this distance as the tuple (d, α) of the distance d between l and the midpoint of s and the angle α between s' and l (see figure 3 (b))[1]. These distance functions fulfill the requirements we state above: Finite and infinite vanishing points are treated in the same way and the distances between points, lines and line segments are independent of their location on the image plane. Note, with this simple approach we disregard the error of a detected line segment and of a potential vanishing point. The modeling of these errors would lead towards a more complex and probabilistic framework.

On the basis of this framework, we can formulate and fill the accumulator space. The intersection points, perhaps at infinity, of all pairs of non-collinear line segments are considered as accumulator cells, i.e. potential vanishing points[2]. Since a vanishing point in the 3D scene is a point at infinity, the corresponding vanishing point in the 2D image cannot lie on a line segment, i.e. between the two endpoints of a line segment, with this vanishing point. Therefore, all potential vanishing points are removed which do not satisfy this condition. In order to fill the accumulator space, we state that a line segment s votes for an accumulator cell a if the distance $d(a, s)$ is below a certain threshold t_a and the vanishing point does not lie on the projected line segment s', which correspond to s. In the search step we are interested in the total vote of an accumulator cell. This vote depends on the length of a line segment[3] as well as on the distance between accepted line segments and the accumulator cell. We define

$$vote(a) = \sum_{\text{all accepted } s \text{ of } a} w_1 \left(1 - \frac{d(a,s)}{t_a}\right) + w_2 \left(\frac{\text{length of } s}{\text{maximal length of } s}\right) \quad (1)$$

[1] We will see later that it is unnecessary to have a scalar as distance measure.
[2] In the following we do not distinguish between an accumulator cell and a vanishing point.
[3] We assume that longer line segments are more reliable than shorter line segments.

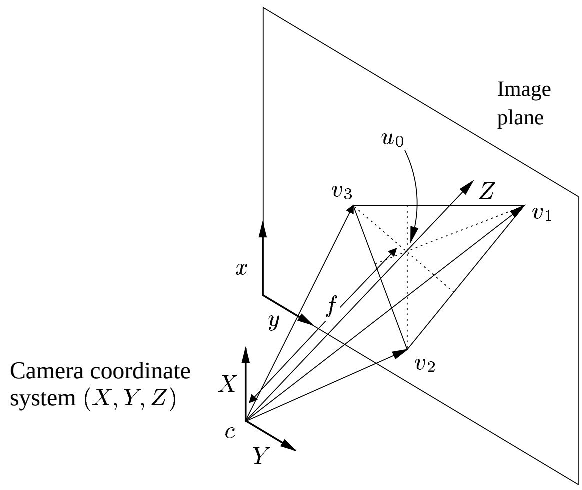


Figure 4: Explanation for the camera geometry and the orthogonal criterion.

as the total vote of an accumulator cell a, where the weights w_1 and w_2 establish this trade off. A brute force version of the accumulator algorithm checks the acceptance of each line segment for each accumulator cell. Afterwards the total vote of each accumulator cell is determined. The computational effort of this algorithm is $O(an) = O(n^3)$ where $a = O(n^2)$ is the amount of accumulator cells and n the amount of line segments.

3.2 Search Step

The task of the search step is to determine the vanishing points, which correspond to the three mutual orthogonal directions of the scene. Due to this constraint on the vanishing points, three different criteria for the vanishing points can be identified: *Orthogonal criterion, camera criterion and vanishing line criterion*. The first two criteria are affiliated with each other and we consider them first.

In [4, 8, 13] was shown that knowledge about the camera geometry can be deduced from the vanishing points of three mutual orthogonal directions. Unfortunately, this knowledge differs in the three different cases, in which none, one or two vanishing points are at infinity. Since these different cases are algebraicly discussed in [8], we illustrate and summarize them here. In order to formulate the orthogonal criterion we have to establish the geometry between the image plane and the camera. We use the same camera model as in [8, 13], which deviates from the general perspective camera in the respect that both image axes are assumed to be orthogonal with the same scale factor. This condition is in general fulfilled for most real world cameras. Therefore, the only unknowns of the camera geometry are the focal length f and the principle point u_0 (see figure 4).

We denote the three vanishing points on the image plane by v_1, v_2 and v_3. Since the vector cv_i from the camera centre c to the vanishing point v_i has the vanishing point v_i, we can formulate the *orthogonal criterion* as:

$$\langle cv_1, cv_2 \rangle = 0, \langle cv_1, cv_3 \rangle = 0 \text{ and } \langle cv_2, cv_3 \rangle = 0 \quad \text{where } \langle \cdot, \cdot \rangle \text{ is the scalar product.}$$

The question for the orthogonal criterion is: Do the vanishing points v_1, v_2 and v_3 satisfy these three equations with reasonable values for u_0 and f, i.e. $f \in (0, \infty)$. We discuss the three different cases with none, one or two vanishing points at infinity separately:

1. Three finite vanishing points v_1, v_2 and v_3:
   The triangle (v_1, v_2, v_3) forms together with the principle point an orthocentric system (see figure 4). Therefore, the intersection point of the heights of this triangle defines the principle point. The size of this triangle defines the focal length uniquely. The orthogonal criterion can be defined as the condition that each angle of this triangle is smaller than 90^o.

2. Two finite vanishing points v_1, v_2 and one infinite vanishing point v_3:
   The principle point lies on the line segment, which is defined by the two endpoints v_1 and v_2. For real world cameras the principle point is more likely positioned in the centre of the image. Therefore, we choose the principle point as the point, which lies on the line segment and is closest to the midpoint of the image. With the determining of the principle point, the focal length is uniquely defined. In this case the orthogonal criterion is defined by the condition that the direction of the infinite vanishing point v_3 is orthogonal to the line defined by v_1 and v_2.

3. One finite vanishing point v_1 and two infinite vanishing points v_2, v_3:
   In this case the principle point is identical with the vanishing point v_1. The focal length cannot be determined. The orthogonal criterion is defined by the condition that the directions of v_1 and v_2 are orthogonal.

We can now specify the *camera criterion*. This criterion is fulfilled when the principle point and the focal length are inside a certain range, in the case they are calculable.

Let us consider the *vanishing line criterion*. Two vanishing points v_1 and v_2 have a vanishing line when not both vanishing points are at infinity (see figure 1). Therefore, a line segment, which lies on the vanishing line, does vote for the two accumulator cells, which correspond to v_1 and v_2. Hence, we define that two accumulator cells fulfill the vanishing line criterion when each line segment, which vote for both accumulator cells, lies on the corresponding vanishing line. In case both vanishing points are at infinity, the two sets of line segments of the corresponding accumulator cells have two be disjunct. With the distance function $d(l, s)$ we can check if a line segment s lies on a vanishing line l. Since $d(l, s)$ returns a tuple (d, α), we check if d and α are below certain thresholds.

With the criteria developed above we can define an algorithm for the search step:

```
Take the accumulator cell a_1 with the highest vote vote(a_1) (see equation 1)
Go through all pairs of accumulator cells (a_i, a_j)
  If the vanishing line criterion is fulfilled for (a_1, a_i), (a_1, a_j) and (a_i, a_j)
    If the orthogonal criterion and camera criterion is fulfilled for (a_1, a_i, a_j)
      Calculate vote = vote(a1) + vote(a_i) + vote(a_j)
Take the accumulator cells a_i, a_j, with the highest vote vote
```

The vanishing points which correspond to the accumulator cells a_1, a_i and a_j represent the three mutual orthogonal directions of the scene. The computational effort of the search step is $O(a^2 n) = O(n^5)$, where $a = O(n^2)$ is the amount of accumulator cells and n the amount of line segments.

4 Experimental results

Figures 5-7 show three examples, in which our method was applied. The threshold t_a in the accumulator step was set to 5^o, the maximal difference between the principle point and the midpoint of the image was set to a third of the image diagonal and the range of the focal length was not specified.

Figures 5-7 demonstrate that the three mutual orthogonal directions of the scene were detected reliably for these images. This was even possible in a cluttered environment (see figure 7) and with a substantial amount of outliers (see figure 6). Figure 5 (a) shows that the accuracy for vanishing point detection is limited. The solid lines of the building and of the street in front of the building are assigned to the same vanishing point. Indeed, the building is rotated approximately about 20^o relative to the street. The dashed lines in figure 7 (a) and solid lines in figure 7 (b), which represent the same line segments in the image, demonstrate the vanishing line criterion.

Let us consider the processing time of the accumulation step and search step on an Ultra Sparc 10. For 77 line segments and 737 accumulator cells the runtime was 3,50 sec for the accumulator step and 3,16 sec for the search step. A different run with 105 line segments and 1552 accumulator cells needed 9,17 sec for the accumulator step and 14,53 sec for the search step. This shows that our method is applicable when no real-time conditions are required.

5 Discussion and future work

A new method for detecting the three mutual orthogonal directions of a man-made environment has been presented. Since real-time performance is not necessary for architectural application, like building reconstruction, a computationally more intensive approach has been chosen. A simple and coherent framework for the accumulation step and search step has been introduced. By using the unbounded image plane as accumulator space, the original distances between vanishing points and line segments are preserved, compared to techniques which transfer the line segments from the image plane into a bounded space. In the search step, criteria for vanishing points of three mutual orthogonal directions have been identified and incorporated. Furthermore, falsely detected vanishing points can be rejected by examine the determined camera parameters of a simplified camera model.

The experiments have shown that the method produced good results, even for images with a cluttered environment and with a substantial amount of outliers. In our future task of building reconstruction we will see, if the accuracy of the detected vanishing points is sufficient. If not, the transition from our simple framework to a more complex and probabilistic framework might be necessary.

Especially buildings have the property that in certain poses not enough – or even none – of the detected line segments specify a search vanishing point. Therefore, we currently extend our method towards an approach which detects all 'visible', mutual orthogonal directions of a scene.

Acknowledgment

We thank S.Carlson and J.-O. Eklundh for their carefully reading of a draft.

References

[1] D. H. Ballard and C. M. Brown: *Computer Vision*. Prentice-Hall, Englewood Cliffs, NJ, 1982.

[2] S. T. Barnard: *Interpreting perspective images*. Artificial Intelligence 21, 1983, pp. 435-462.

[3] B. Brillault-O'Mahony: *New Method for Vanishing Point Detection*. In Computer Vision, Graphics, and Image Processing 54(2), 1991, pp. 289-300.

[4] B. Caprile and V. Torre: *Using Vanishing Points for Camera Calibration*. International Journal of Computer Vision 4, 1990, pp. 127-139.

[5] A. Criminisi, I. Reid and A. Zisserman: *Single View Metrology*. Proceedings of the 11th International Conference on Computer Vision (ICCV'99), September 1999, Kerkyra, Greece, pp. 434-441.

[6] J. M. Coughlan and A. L. Yuille: *Manhattan World: Compass Direction from Single Image by Bayesian Inference*. Proceedings of the 11th International Conference on Computer Vision (ICCV'99), September 1999, Kerkyra, Greece, pp. 941-947.

[7] F. A. van den Heuvel: *Vanishing point detection for architectural photogrammetry*. International Archives of Photogrammetry and Remote Sensing, Vol. XXXII part 5, 1998, pp. 652-659.

[8] D. Liebowitz and A. Zisserman: *Combining Scene and Auto-calibration Constraints*. Proceedings of the 11th International Conference on Computer Vision (ICCV'99), September 1999, Kerkyra, Greece, pp. 293-300.

[9] E. Lutton, H. Maître and J. Lopez-Krahe: *Contribution to the determination of vanishing points using Hough transform*. IEEE Transaction on Pattern Analysis and Machine Intelligence 16(4), 1994, pp. 430-438.

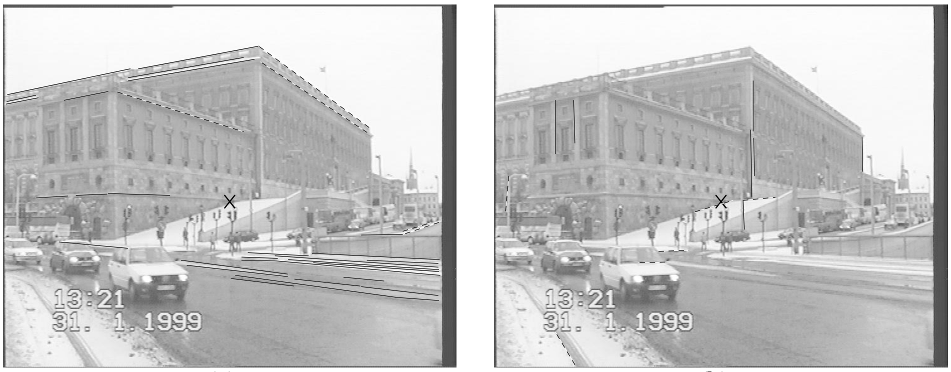
[10] M. J. Magee and J. K. Aggarwal: *Determining Vanishing Points from Perspective Images*. Computer Vision, Graphics and Image Processing 26, 1984, pp. 256-267.

[11] L. Quan and R. Mohr: *Determining perspective structures using hierarchical Hough transform*. Pattern Recognition Letters 9, 1989, pp. 279-286.

[12] M. Straforini, C. Coelho and M. Campani: *Extraction of vanishing points from images of indoor and outdoor scenes*. Image and Vision Computing, Vol. 11(2), 1993, pp. 91-99.

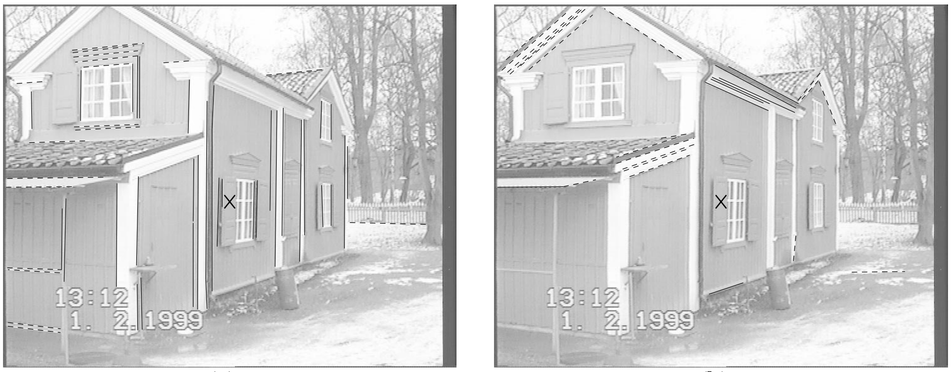
[13] D. Svedberg and S. Carlsson: *Calibration, Pose and Novel Views from Single Images of Constrained Scenes*. Proceedings of the 11th Scandinavian Conference on Image Analysis (SCIA'99), June 1999, Kangerlussuaq, Greenland, pp. 111-117.

[14] T. Tuytelaars, L. Van Gool, M. Proesmans and T. Moons: *The Cascaded Hough Transform as an Aid in Aerial Image Interpretation*. Proceedings of the 6th International Conference on Computer Vision (ICCV'98), January 1998, Bombay, India, pp. 67-72.



(a) (b)

Figure 5: An image of the royal castle of Stockholm. The two different types of lines in (a) and the solid lines in (b) display the vanishing points of the three mutual orthogonal directions. The dashed lines in (b) represent the remaining line segments, which were not assigned to one of the three vanishing points. The detected principle point of the camera is drawn as a cross.



(a) (b)

Figure 6: An image of a residential building. The line markings are as in figure 5.

(a) (b)

Figure 7: An image of a residential building, whereat the line markings are as in figure 5.