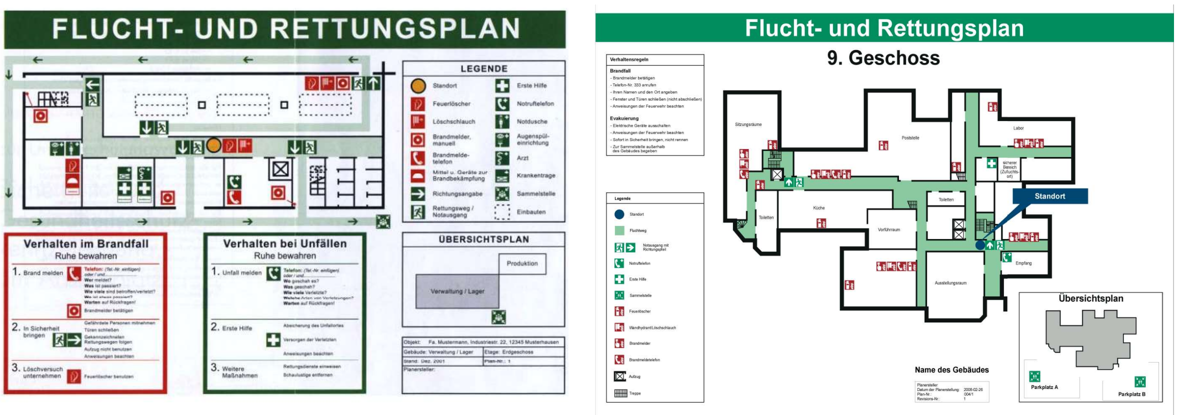
which were automatically detected in the provided plan image. The final result is a CAD file consisting of layers containing architectural elements like windows, doors and furniture.

In contrast to these related approaches dealing with CAD files or scanned plans, the approach developed in the course of this thesis uses photographed evacuation plans together with a model of the building's external shell to deduce geo-referenced 2D and 3D models. Obviously, one of the main difficulties is the use of photographs in contrast to high resolution scans with complete control over the employed lighting conditions. This poses problems such as brightness differences throughout one image and between different images, resulting also in differing colour reproduction. While the colour reproduction is not an issue when dealing with CAD models and occlusions are minimal, the automatic analysis of evacuation plans suffers from occlusions caused by coloured symbols depicting evacuation paths or safety equipment. Many of the aforementioned approaches use symbol detection for the identification of door arcs and similar features. The symbols used in evacuation plans, however, can generally only be detected using the color information which generally is deteriorated by the lighting conditions. Finally, most approaches involve at least some kind of manual interaction, be it for the cleansing of the binary image, text removal or the solution of the scale problem. Here, the latter problem is solved automatically by matching the indoor model to a model of the building's external shell.

4.2 Evacuation plans

For the developed reconstruction method for building interiors, photographed evacuation plans are used as one of the data sources. The most important reasons to choose these plans consist 1) in their ubiquitousness, and 2) their design's degree of standardization. In fact, their ubiquitousness is part of their standardization as the International Organization for Standardization (ISO) defines evacuation plans as "essential part of the safety equipment of a physical structure" in DIN ISO 23601. DIN ISO 23601 is the international norm which aims at the harmonization of the various laws and bylaws regulating the design of such plans on national levels. In Germany, for example, the laws and bylaws ensuring the availability of evacuation plans in all publicly used buildings as well as at all workplaces comprise the „Berufsgenossenschaftliche Vorschrift für Sicherheit und Gesundheit bei der Arbeit (BGV-A8)" and the DIN 4844-3.



(a) (b)

Figure 4.4: Evacuation plans according to a) BGV A8 and b) DIN ISO 23601

Figure 4.4 shows the plan specimens - from BGV-A8 and DIN ISO 23601 respectively - which serve as illustration of the design principles. These design principles range from the definition of specific element sizes and compulsory elements to rules for the plan's location and materials to use for its production. Table 4.1 contains the most important design principles of both norms which are usable in the context of an automated analysis. This comparison exhibits that the norms agree in many standardized features supporting an automated analysis, as shown in the last column. For example, the colour specifications can be used in the context of image enhancement (white background reconstruction and colour correction, see section 4.4.3) and binarisation (facilitated differentiation between background and ground plan lines), or in the detection of coloured symbol regions. Minimum size constraints for the symbols and the plan itself as well as the compulsory use of standardised safety signs (square/rectangular) constrain the size of symbols in the image.

<table>
<tr><th>Feature</th><th>DIN 4844-3:2003</th><th>DIN ISO 23601:2010</th><th>Usage in automated analysis</th></tr>
<tr><td>use of colours</td><td colspan="2">yes</td><td>detection of symbol areas, occlusions</td></tr>
<tr><td>background colour</td><td colspan="2">white</td><td>color correction, facilitated binarisation</td></tr>
<tr><td>ground plan colour</td><td colspan="2">black</td><td>facilitated binarisation</td></tr>
<tr><td>colour evacuation routes</td><td>light green/darker green (stairs)</td><td>arrows green; evacuation routes lighter green</td><td>detection of symbol areas, occlusions; information about the direction of staircases; information about the location of doors</td></tr>
<tr><td>minimum plan size</td><td colspan="2">DIN A3</td><td rowspan="2">size constraints for the symbol detection</td></tr>
<tr><td>symbol size</td><td>preferable 10mm / “you are here” at least 10mm</td><td>minimum height 7mm</td></tr>
<tr><td>standardised symbols</td><td>BGV A8 and DIN 4844-2</td><td>ISO 7010</td><td>templates for template matching based approach</td></tr>
<tr><td>correct positional arrangement</td><td colspan="2">yes</td><td>computation of the initial orientation for positioning (see chapter 5)</td></tr>
</table>

Table 4.1: Overview over design regulations in DIN 4844-3:2003 and DIN ISO 23601:2010

Despite such detailed (and in theory) internationally applicable design rules, the reverse-engineering approach was designed to be sufficiently general instead of sticking too closely to the standardization documents. This results from, first, the fact that basing only on the German regulations would theoretically constrain the applicability of the approach to only German plans. Second, the regulations leave space for interpretation (stating in the DIN 4844 that at least “all plans inside one facility have to be in the same layout”) and therefore plans differ to some extend. Third, the DIN ISO 23601, as a relatively young international standard, may not be followed everywhere yet. The plans found during data acquisition (in hotels and public buildings in various countries) support these assumptions. However, the data acquisition showed that the rules common to DIN 4844 and DIN ISO 23601 - i.e. the use of coloured symbols and black ground plans on white backgrounds - are used in the vast majority of plans.

The use of photographed evacuation plans for reconstruction can be seen in parallel to aerial images released by their owners being employed as base data in the OpenStreetMap modeling strategy. However, in contrast to those images, the legal situation is not as clear in the case of photographed evacuation plans. Thus, when proposing a manual modelling strategy for OpenStreetMap similar to the one presented in the

OpenStreetMap Wiki[2] in the OpenStreetMap mailing list[3], this caused a certain amount of discussion. Most importantly, references to the Berne convention for the Protection of Literary and Artistic Works were raised. This convention states - in short - that all works[4] should be copyrighted in all signing countries for at least 50 years after the author's death or, if the author is unknown, for 50 years after the first publication. Replies contained that a) not every country has signed the Berne convention and b) modeling from such a photograph instead of publishing the image itself is not covered by the convention. However, a building's interior design itself is creative work covered by the Architectural Works Copyright Act.

This shows that this topic seems to live in a legal grey zone, an opinion which is further supported by statements made among others by Marcus Götz, the author of the article in the OpenStreetMap Wiki (see above), during the Q&A session after his FOSSGIS 2012 talk[5]. Following the argumentation of a lawyer commissioned by him, the design of the evacuation plan is protected, however, the content is not. Similar opinions as mentioned above are expressed: the photograph might not be suitable for presentation, but the derived model might be. Furthermore, the evacuation plan is publicly available and thus not protected. However, conflicts with local regulations given by the building owner or "house rules" might arise.

4.3 Overview

The processing pipeline from the photographed evacuation plan to a final 2D or even 3D model is depicted in figure 4.5. A significant part of it consists of pre-processing steps needed to overcome flaws of the image e.g. perspective distortions and an adulterated colour reproduction (described in section 4.4). The pre-processed image is then binarised (section 4.7) and matched to an available model of the building's external shell in order to geo-reference and - more importantly - scale the model (section 4.5). The symbol areas detected by colour or shape constraints and template matching (section 4.6) are bridged, resulting in a final 2D model (section 4.8). If the plan contains stairs which can be detected, this 2D model can be extruded to a 3D model (section 4.9).

Some of the methods described here were developed in joint work with a Diploma Thesis candidate supervised by the author of this thesis (Otto, 2014). While the Diploma Thesis concentrated on plans with a well-known design, the other approaches aim at generalising the process in order to allow for the reconstruction of arbitrary evacuation plans.

[2] `http://wiki.openstreetmap.org/wiki/IndoorOSM` (last visited 2nd February 2016)

[3] `https://lists.openstreetmap.org/pipermail/talk/2011-September/060227.html` (last visited 2nd February 2016)

[4] except photographic and cinematographic works for which different regulations where stated

[5] a recording can be found under `http://db0smg.afug.uni-goettingen.de/~fossgis/FOSSGIS2012-468-de-indoor.mp4`

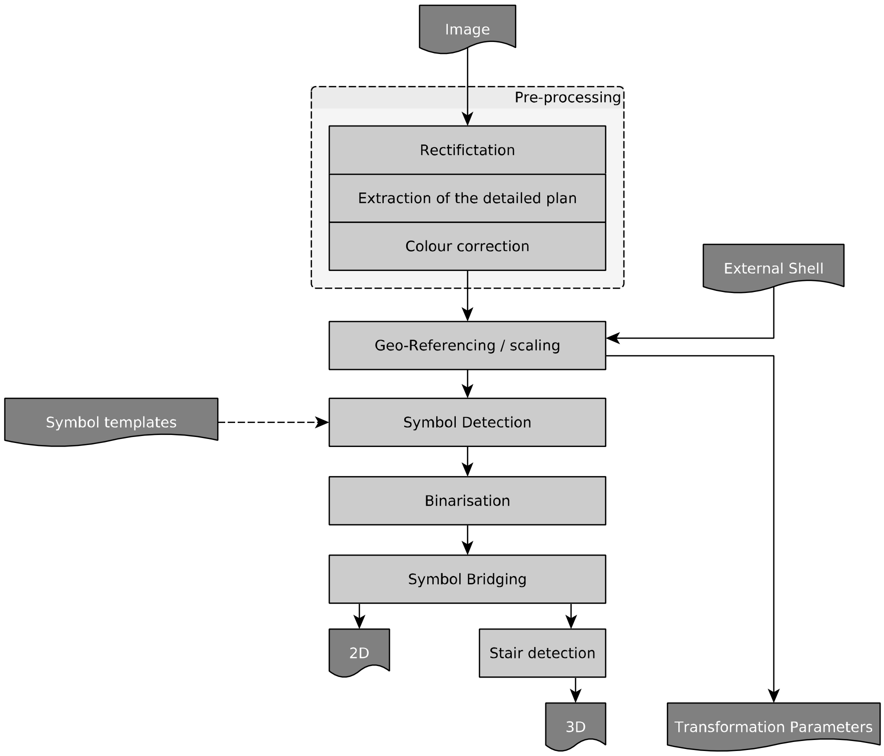


Figure 4.5: Flow chart of the processing pipeline from photographed evacuation plan to 2D/3D model

4.4 Image pre-processing

In contrast to the methods described in section 4.1 which base on scanned, high-resolution images, the photographs used here suffer from brightness and colour differences as well as reflections. Due to varying lighting conditions, this is not only true for images taken with different cameras, but also for different images taken with the same camera and for different regions inside a single image. Furthermore, in order to avoid reflections deteriorating the image quality, it may need to be taken at an oblique angle instead of perpendicularly. Section 4.4.1 covers a method to overcome the latter problem while sections 4.4.2 and 4.4.3 describe steps carried out to repair colour and brightness differences throughout the image.

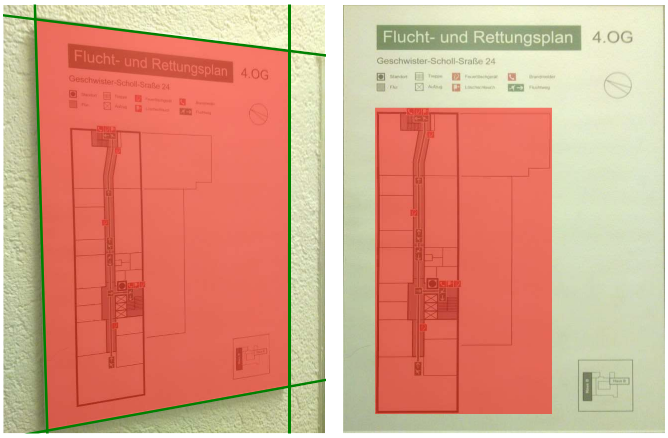
4.4.1 Image rectification and detection of the detailed plan

If the plan was photographed at an oblique angle, the image has to be rectified in order to be usable for the reconstruction. The perspective transformation parameters for the rectification can be recovered if parts of the frame of the paper the plan is printed on are visible in the image, similarly to the whiteboard scanning method presented by Zhang and He (2007).

To this end, the following steps are carried out:

1. In order to reduce image noise and remove other small structures disturbing the further analysis, the image is blurred heavily, and a bilateral filter is applied

2. edges in the blurred image are detected using the Canny operator (Canny, 1986)



(a) (b)

Figure 4.6: Rectification of the image and detection of the detailed plan

3. dominant lines are detected in the edge image using the Hough transformation and merged using the buffering operation presented by Kada (2007) in the context of cartographic building generalisation (described in more detail in section 4.5)

4. the topmost, lowermost, leftmost and rightmost dominant lines are selected and intersected

5. from the intersection points, the perspective transformation can be computed (see Appendix A on page 137), setting the image corners as target points. This will result in small distortions if the image's aspect ratio does not match the plan's aspect ratio. However, these distortions will be removed by the estimation of an affine transformation during the geo-referencing operation (see section 4.5).

In figure 4.6a) the detected and merged lines are shown, while figure 4.6b) depicts the rectified and cropped image. Furthermore, the image can be cropped to the detailed plan which will be used further in the processing pipeline. This is done using the following two steps: 1) binarise the rectified and cropped image using adaptive thresholding, 2) detect connected components and select the biggest contour (using Suzuki et al., 1985). Based on this contour's bounding box, the sought-after detailed plan is selected (see red rectangle in figure 4.6b).

4.4.2 Background normalisation

To overcome the brightness differences in a single image, Otto (2014) implemented a two-step process consisting of retroactive white balancing and the normalization of the image's background intensity. The idea for the white balancing method is taken from the image editing software "The GIMP"[6]. It consists of analysing both ends of the image's grey value histogram and discarding pixel colours which only a certain percentage of pixels use (representing dust or similar disturbances). While in GIMP a threshold of 0.05% is used, Otto (2014) sets it to 5%. This is followed by a histogram stretching operation which restores maximized contrast by ensuring the use of the full intensity interval of [0;255].

[6] `http://docs.gimp.org/2.8/en/gimp-layer-white-balance.html` (last visited 2nd February 2016)

The normalization of the image's background intensity bases on the idea of reconstructing the background brightness by removing all the foreground structures[7]. The foreground removal is implemented as a morphological opening of the grey scale image with a big structuring element[8]. Subsequently, the normalized image can be computed by dividing the grey scale image by the background image. The disadvantage of this approach is found in the fact that it enables only a correction of the grey scale image with the aim of allowing for a more robust binarisation. However, an overall correction of the image's colours is not feasible.

4.4.3 White background reconstruction and colour correction

In order to further improve the colour-based symbol detection an approach for the correction of the colour image was developed. This colour correction approach bases on the design guidelines presented in section 4.2, exploiting the white background which can be expected in a photograph of an evacuation plan. The colour correction is carried out after converting the image to the CIE L*a*b*[9] colour space. This colour space is based on the human colour perception, representing colours in a three-dimensional coordinate system whose axes are the colour's Lightness (L*) as well as a chromacity value with respect to the red/green axis (a* axis) and the yellow/blue axis (b*). In the original definition, the L* channel's domain is defined as [-100;100] and the a*/b* channels vary between -128 and +127. However, in order to enable their storage in 8 bit images, they are often scaled to [0;255], resulting in the white and black point being at (255,128,128) respectively (0,128,128). Due to these characteristics, a correction of the L* channel affects all colours present in the image.

Figure 4.7a) shows one of the expected problems of the image's quality. The cyan line represents the L* channel values of a single image column close to the border of the imagewhich is assumed to be completely white (see figure 4.8a)) before any correction. A brightness decrease towards the ends of the column (i.e. the image border) is clearly visible, caused by a lens limitation commonly called vignetting. Depending on the lighting conditions, this defect induced by the lens can be overlaid by a linear trend if the light is distributed inequally throughout the image (see dashed red line in figure 4.7a)).

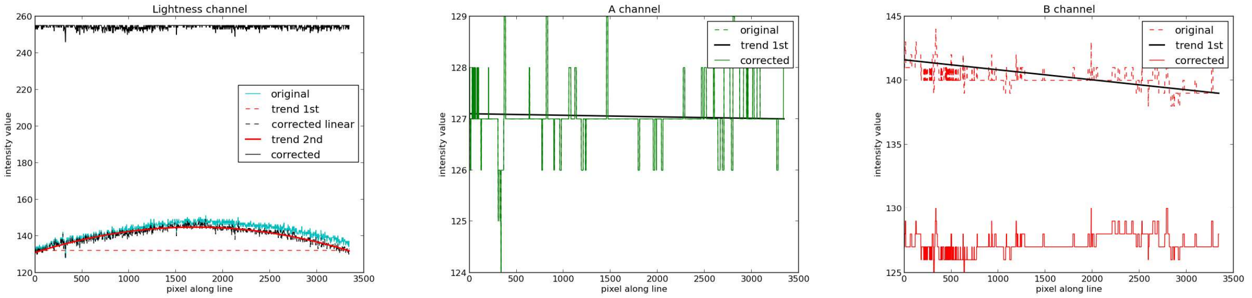
For the correction of the L* channel image, first the linear trend is estimated using the 10 first and 10 last pixels of the line. The result of the linear trend's removal from the original data (figure 4.7a), cyan line) is depicted as a black dashed line. In a second step, the offset parameter from the estimated linear polynomial is used together with a threshold of 25 to select L* channel values which allegedly represent the white background in the image. Finally, a second degree polynomial is estimated using the selected values (figure 4.7a), red solid line). The subtraction of this trend from the data and a simultaneous translation of the mean to 255 results in the final data, depicted in figure 4.7a) by the solid black line.

The partitioning into lightness and chromacity channels delivered by the conversion to the CIE L*a*b* colour space ensures that the brightness differences are completely corrected by the L* channel correction. However, the chromacity channels may contain slight colour changes throughout the image and, more importantly, tints which are reflected in a translation of the white point away from its nominal value of 128

[7] `http://dsp.stackexchange.com/questions/1932/what-are-the-best-algorithms-for-document-image-thresholding-in-this-example` (last visited 2nd February 2016)

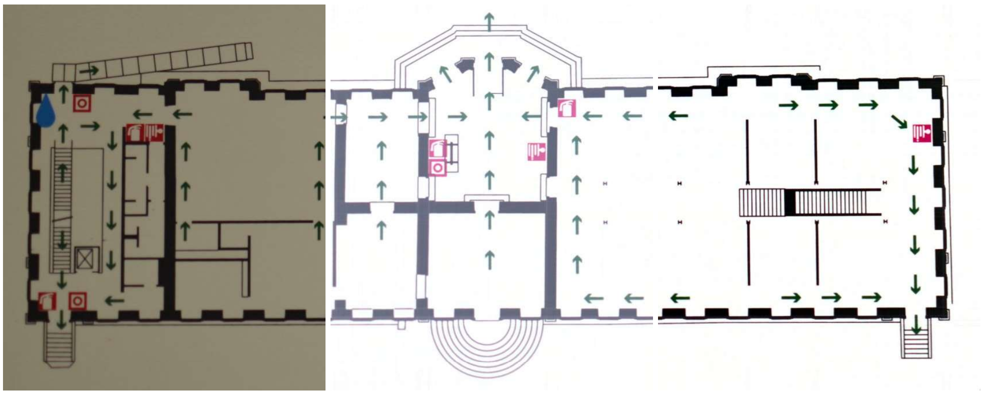
[8] with a size of e.g. 1/3 of the shorter image edge

[9] DIN EN ISO 11664-4



(a) (b) (c)

Figure 4.7: Corrections along one image row for white background reconstruction and colour correction



(a) (b) (c)

Figure 4.8: Original (a), colour corrected (b) and histogram stretched (c) images

(see figure 4.7c) in comparison to figure 4.7b)). Thus, for the a* and b* channels only the linear trend is estimated and corrected in combination with a translation of the mean value to 128. The corrected image is converted back to the RGB colour space.

As visible in figure 4.8b), the thusly enhanced image lacks contrast. To overcome this flaw, it is treated with the white balancing/histogram stretching method described before (using a threshold of 0.05% for the pixels to be discarded), resulting in the image depicted in 4.8c).

4.5 Geo-referencing and scaling

A common problem of the related approaches presented in section 4.1 is the definition of the coordinate transformation between the resulting image coordinates and a metric reference system. Without its definition in metric coordinates, the resulting model can serve visualization purposes at most. In the related work, the metric scale is mostly derived using human interaction (e.g. definition of one or more edge lengths). While scaling the model is sufficient for applications like indoor positioning and navigation as well as interior architecture, geo-referencing it in a world coordinate system allows for a seamless navigation between outdoor and indoor areas as well as the co-registration of different floors.

Additionally, the metric scale is needed for the approach presented here in order to define metric thresholds for the symbol bridging operation. Therefore, an automatic geo-referencing method was developed which

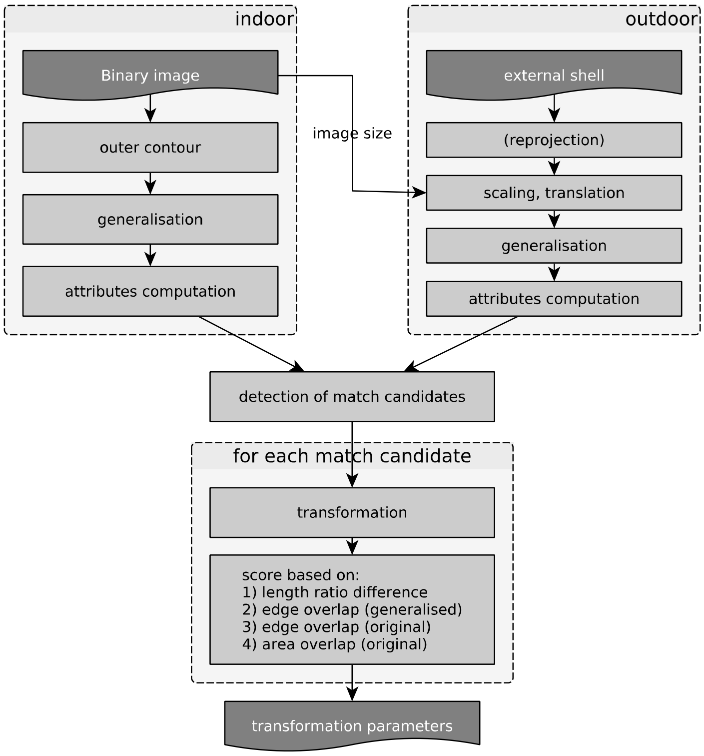


Figure 4.9: Flow chart depicting the geo-referencing process

makes use of contour matching between the model depicted in the evacuation plan and a model of the building's external shell available in a world coordinate system. The flow chart in figure 4.9 depicts the whole process which is described in depth in the following sections.

4.5.1 Preparation of the contours

The outer contour of the floor plan depicted in the evacuation plan - in the following called *indoor contour* - is detected in a binary image derived using adaptive thresholding on the pre-processed image. Small gaps are removed using morphological closing, followed by the detection of the contours contained in the image using the topological structural analysis approach described in Suzuki et al. (1985). If the image was cropped to show only the detailed plan, the sought-after indoor contour is found as the largest contour by area.

The external shell, on the other hand, may be given to the process as a ground plan or 3D model from an available city model, or, alternatively, can be taken from OpenStreetMap. Depending on the source, the first step is the projection of the polygons' coordinates to a projected coordinate system (e.g. Gauß-Krüger or UTM). Subsequently, the centroid of the model is subtracted and the model is scaled to fit into an image twice as large as the longest edge of the evacuation plan image by applying the following two equations:

$$c_{im} = \frac{X_{proj} - \overline{X_{proj}}}{s} + \max(r_{im}, c_{im}) \tag{4.5.1}$$

$$r_{im} = \frac{\max(Y_{proj} - \overline{Y_{proj}})}{s} - \frac{Y_{proj} - \overline{Y_{proj}}}{s} + \frac{\max(r_{im}, c_{im})}{2} \tag{4.5.2}$$

where X_{proj}, Y_{proj} are the coordinates of the external shell in a projected coordinate system, r_{im}, c_{im} are the row/column image coordinates, max(r_{im}, c_{im}) is the maximum of the image dimensions, $\overline{X_{proj}}, \overline{Y_{proj}}$ are the coordinates of the external shell's centroid and $s = \max(\max(X_{proj}) - \min(X_{proj}), \max(Y_{proj}) - \min(Y_{proj}))$ $/\max(r_{im}, c_{im})$ is the approximate scale derived from the comparison of the longer edges of both contours' bounding boxes. This scaling ensures that the same parameters can be used in the following cartographic generalisation for both the indoor and the outdoor contours. The parameters of equations 4.5.1 and 4.5.2 together with the affine transformation estimated in the last step of the process define the reverse transformation from image coordinates to the world coordinate system.

4.5.2 Cartographic generalisation

Due to their differing scales, in most instances both contours will show quite different levels of detail, a fact which could prevent the applicability of the following contour matching approach. In order to overcome this problem, they are simplified using a cartographic generalisation method for building ground plans similar to the one presented in Kada (2007). Kada (2007)'s shape-preserving generalisation method for 3D building models bases on half-space modeling and cell decomposition. Here, only the ground plan simplification component is used, based on the ground plan edges and not the faces connected to them.

The essence of the approach is the aggregation of similarly oriented edges of the contour using a buffering operation. To this end, in a first stage the buffers are constructed from the edges using a normal vector based line representation (normal vector plus minimum and maximum distance) and the edges' weight deduced from their length. Then, the respective strongest buffer is determined by sorting the buffers by their weight and checking for other buffers which are mergeable (fulfilling the angle threshold and not causing the buffer to grow bigger than the distance threshold) or includeable (buffers not fulfilling the angle threshold and not causing the buffer to grow bigger than the distance threshold). The buffer merging operation causes the recomputation of the buffer's normal vector as well as its bounds, while the including operation only affects the bounds. This process is repeated in a loop and produces the respective strongest buffer for which the distance parameter is recomputed as the weighted mean distance of all buffers which were merged.

In a second stage, the lines derived from the found buffers are intersected, producing a cell decomposition representation of the sought-after simplified polygon. From the set of cells the ones which fulfil a threshold for the area overlap with the original polygon are selected and the final generalised model is constructed as the union of these cells.

While strongly simplifying the building model, this approach manages to retain the model's overall structure very well. The choice to compute the weighed mean distance results in simplified models which are optimized towards *area trueness* in comparison to the original models (Kada et al., 2009a). While this is a preferable feature in many cases, often the simplified models are requested to fulfil characteristics like a maintained common façade line or - more generally - *trueness to the nodes* of the original model. Thus, an extension to Kada (2007)'s approach was developed which aims at adjusting the generalised model to the original ground plan (Peter et al., 2008; Kada et al., 2009a; Peter, 2009). The extension consists of two steps: 1) the adjustment of the generalised ground plan to the strongest line constructible from each buffer and 2) the 3D model's adaption to the adjusted ground plan. The ground plan adjustment is done by employing two more strict thresholds for distances and angles in order to build the strongest constructible line instead of computing the line representing each buffer as the weighed mean.

4.5.3 Contour matching

With the availability of the generalised indoor and outdoor contours, the transformation parameters can be determined by matching the contours. Contour matching approaches for general polygons are available in related work, put to use for example in automatic puzzle solving or reconstruction of disrupted documents (Kong and Kimia, 2001). Veltkamp (2001) describes more applications for shape matching and the cores of various algorithms, their similarity measures, as well as the voting schemes which are used to compute the final similarity measure.

In the case of the generalised indoor and outdoor contours the problem is less general, as right angles will be most prominent and the polygons will supposedly be very similar after generalisation. Thus, a custom contour matching method was developed which leverages the per-node length ratio of the adjacent edges as well as the angle enclosed by them. Based on the length ratio difference and the angle difference, nodes fulfilling very loose thresholds[10] are chosen as match candidates.

For each of the match candidate nodes, the indoor and outdoor contour nodes themselves as well as their respective two neighbours are used to determine the parameters of an affine transformation (see Appendix A on page 137) which is used to transform the original and the generalised indoor contours[11]. Based on the transformed generalised indoor contour and the generalised outdoor contour, the following scoring parameters are computed: a) the mean length ratio difference of all nodes' adjacent edges and b) the edge overlap, determined using a buffer around the outdoor contour which has the size of the generalisation distance (as used in the quality assessment method presented in Filippovska (2012)). Additionally, c) the edge overlap and d) the area overlap between the original outdoor contour and the transformed original indoor contour are computed (see figure 4.10).

The best set of transformation parameters is chosen based on these scoring parameters, firstly scoring by a lower mean length ratio difference, followed by the decision based on a higher edge overlap. In the case of symmetric generalised models, however, these parameters may deliver an ambiguous result. Thus, the edge overlap as well as the area overlap between the original contours are also taken into account (see figure 4.10).

As an improvement, additional homologous points in the transformed indoor contour and the outdoor contour can be identified. These can be used either to improve the estimation of the affine transformation or to estimate the perspective transformation instead. However, this is generally not necessary, as the perspective distortions are removed during pre-processing already (see section 4.4).

The resulting transformation parameters may also be used to transform the photographed plan. Thus, it can serve as a backdrop to visualize positions and pedestrian traces without explicit reconstruction of a model.

[10] the threshold for the length ratio difference was chosen as 0.5, the angle difference threshold was set to 10° and kept for all data sets which were tested

[11] instead of a similarity transformation the affine transformation is needed, because the evacuation plan might be differently scaled along the two coordinate axes

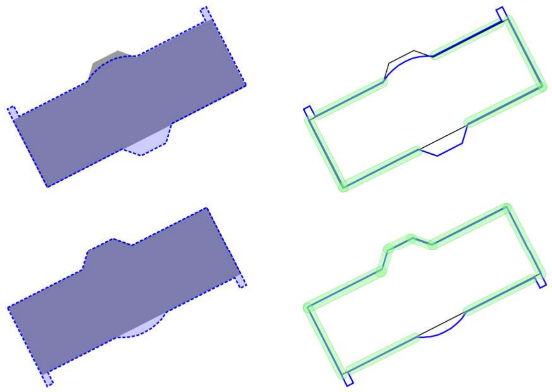


Figure 4.10: Symmetric generalisation result: decision based on area overlap (left) and edge overlap (right) between the original contours

4.6 Symbol detection

In contrast to the scanned architectural plans which serve as input to most of the methods described in section 4.1, photographed evacuation plans contain symbols depicting the location of emergency equipment (e.g. fire extinguishers, water hoses, communication facilities) or evacuation routes. As emergency equipment often is installed at walls, its symbols are likely to occlude parts of walls, while evacuation route symbols often are drawn over doors or stairs. Thus, in order to reconstruct a complete model, the symbols have to be detected and bridged. Apart from this most important reason, the symbols should be detected in order to represent the information as well as meta-information they contain in the final model. One example for meta-information is the down[12] direction of a flight of stairs which is covered by an evacuation route symbol, as evacuation routes are much more likely to lead to lower levels of the building than to the roof. Thus, the detection of symbols is a crucial part of the evacuation plan analysis.

4.6.1 Symbol detection using contour detection and template matching

The symbol detection method described in Otto (2014) bases on the facts a) that the input plan's layout is known, including the symbols which are available as templates, and b) that the symbols are surrounded by a white padding which separates them from the structures they occlude. The method consists of the following steps:

1. The first step of Otto (2014)'s method aims at the detection of the predominant symbol size in the image of the plan. As the layout of the plans used in the experiments closely follows the DIN 4844-3, the symbology consists of squares (emergency equipment, evacuation route arrows) and rectangles with an aspect ratio of 1:2 (evacuation route arrows with running person). The red signal colour of the emergency equipment symbols can be detected robustly by converting the image to the CIE L*a*b* colour space (see also section 4.4) and analysing the a* channel with a simple threshold. Using connected component analysis (Gonzales et al., 2004) with the resulting binary image, the red symbols' contours can be detected which are filtered by their squareness as well as by a size threshold derived from the size constraints stated in the norm. The predominant symbol size results as the median of all remaining bounding boxes' edge lenghts.

[12] if the floor is in an overground level

2. The background normalised image (see section 4.4) is binarised using adaptive thresholding. Using a second connected component analysis, all contours (rooms, stairs, symbols) can be extracted from the binary image. Here, the aforementioned white padding around symbols is leveraged, as it results in all symbols being found as separate contours. Symbol candidate contours are identified using the symbol size computed before by filtering the contours, only allowing those with sizes and aspect ratios *(symbol size):(symbol size)*, *(symbol size):(2 x symbol size)* or *(2 x symbol size):(symbol size)*. Further constraints include minimum length constraints for both the short and the long bounding box edges which depend on the symbol size, as well as a constraint excluding round contours.

3. As the templates for the symbols are available e.g. from the norm documents or extracted from one plan's legend, in theory, an identification of the symbol's meaning would be possible using template matching. In practice, however, the detection depends on thresholds which would have to be defined for each photographed plan separately, as Otto (2014) states. In the thesis, this approach is demonstrated for the "you are here" symbol only, which is very distinct from the other symbols. The fact that the regions of interest were detected before and their dimensions are known helps robustifying the method, as the template can be scaled to the correct size before matching as well as only the regions of interest have to be analysed.

This method, however, does not allow for a detection of the evacuation routes which are depicted as simple green lines between the evacuation route arrow symbols. Otto (2014) describes three different ways to detect this missing information: 1) detection of the remaining green areas and filtering using a size constraint, 2) detection of isolated contours between evacuation route arrow symbols in the direction of the arrows or 3) skeletonisation and vectorisation of the binary image cleaned by the symbol areas and detection of isolated line strings.

4.6.2 Symbol detection using colour segmentation

If the two prerequisites mentioned at the beginning of the previous sub-section are not met, the method developed in Otto (2014) will fail. For one, the symbols may not be rectangles but have arbitrary shapes (like arrows etc.). Secondly, they may be connected to wall structures, which renders their detection based on the contours' shapes in the binary image impossible.

Thus, a more general method based solely on the symbols' colours was developed. To this end, a colour segmentation method called Color Structure Code, presented in Priese and Sturm (2003) is used. Due to its hierarchical nature, this region growing approach uses local and global information to split the colour image into similarly coloured regions. Its result is represented by a label image and an image containing the similarly-coloured regions filled with their mean colour. The colour detection based on these regions was found to be more stable than based on the analysis of single pixels.

The colour classification is carried out on the CIE L*a*b* representation of the region image which results from the Color Structure Code as follows:

1. Detect the maximum and minimum values along the a* and b* axes, map the values to $[0;1]$ by subtracting 128 and dividing by *(max-128)* (positive values) or *(128-min)* (negative values).

2. Red and green are assumed to be existing in all plans in order to signalize emergency equipment and evacuation route symbols. The existence of blue and yellow is checked by $min(b^*) < min(a^*)$ (blue) and $max(b^*) > max(a^*)$ (yellow).

3. The final colour classification is done by means of the following thresholds (in this order): $a^* > 0.7$ and not $b^* < 0.4$ (red), $a^* < 0.4$ (green), $b^* < 0.4$ (blue) and $b > 0.6$ (yellow).

While this approach is more generally applicable than the template-based method, it is less robust (as building on less external information). Thus, coloured regions might be left undetected and be interpreted as structure to be included in the final model. Although these errors in the model must be corrected by manual interaction, the method nonetheless saves digitization time in all correctly vectorised parts of the model.

4.7 Binarisation

Using the images resulting from his aforementioned background normalisation method, Otto (2014) reports that the use of the normalised binarisation method presented by Sauvola and Pietikäinen (2000) delivers the best results for the layout of the plans used in his study. This method was developed for the binarisation of scanned documents and is optimised towards slight brightness differences in the image and black letters on white background. To this end, a threshold is selected per pixel based on the mean value as well as the standard deviation in a local neighbourhood.

In addition to Otto (2014)'s approach, a custom binarisation method was developed, motivated by two reasons: 1) evacuation plans may contain areas shaded in grey which depict hallways or other regions of interest, 2) light green areas like the evacuation routes which can be seen in figure 4.4b) are not detected by the symbol detection method described before. Depending on the neighbourhood size parameter chosen for the adaptive thresholding method, both of these may deteriorate the binarisation result by areas falsly categorized as foreground.

Therefore, a global thresholding method was developed which takes into account that - in addtion to white - a second colour may exist in the image which must be categorized as background. To this end, the following steps are carried out:

1. Set the detected symbol regions to white in the grey scale image.

2. Compute the histogram, smooth it.

3. Detect local extrema in the smoothed histogram, identify the first (black) and last (white) distinct maxima.

4. Search for a distinct maximum between the black and white values

   a) if such a maximum exists use the minimum before as the threshold for the binarisation, or

   b) use a fixed threshold of 200.

4.8 Symbol bridging and 2D modelling

In order to produce a complete final model, structural information that was occluded by the symbol regions has to be restored. To this end, the scale factor derived during the geo-referencing step is used to enable the definition of metric thresholds.

The following process was implemented in order to reach this goal:

1. The binary image is cleaned from the symbol regions and the cleaned binary image's skeleton is computed using the iterative thinning method presented by Zhang and Suen (1984). The iterative thinning method was chosen, as - opposed to distance transformation based methods (like Felzenszwalb and Huttenlocher, 2004) - it tends to produce less unwanted end point edges at sharp corners.
   The skeleton image represents the topological relations in the binary image as branch nodes (pixels with more than two filled neighbouring pixels) and end nodes (pixels with exactly one filled neighbouring pixel). Starting at the branch and end nodes and following the skeleton in between them, the skeleton is vectorized, resulting in an incomplete 2D vector model of the floor plan. The end nodes of this vector model serve as candidate starting points for the symbol bridging process.

2. Short edges with a length of less then 10 centimetres ending in two end points are very likely binarised and vectorised image noise and can be deleted.

3. While the vectorized image skeleton represents the topological relations very well, geometric precision may be lost during skeletonisation and vectorisation, especially at short end point edges. In order to restore perpendicularity, the adjacent edges' orientations of all end point edges shorter than 1 metre are analysed. Subsequently, the end point edge is rotated to be perpendicular to the predominant orientations of the neighbouring edges.

4. All end point edges which are longer than 10 centimetres are prolonged until either structure in the cleaned binary image or the image border is hit.

5. Intersections of the prolonged edges with the symbol contours are computed. Depending on the existence of intersections with symbol contours, the prolonged edges are treated differently:

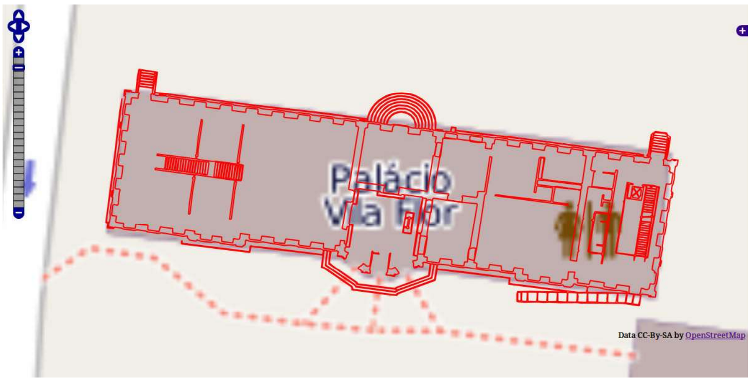
   a) If the prolonged edge does not intersect with any symbol contour, it is selected to be included in the final model if the prolongation is shorter than 1 metre. This is motivated by the fact, that such edges can either close door openings (which are narrower than the standard door width of 0.8 metres) or small gaps in the model which are caused by binarisation errors.

   b) If the prolonged edge intersects with a symbol contour, its intersections with the other prolonged edges are computed. The parts which are entitled to be included into the final model are chosen based on the principle "longer edge stops shorter edge".

6. The final 2D model can be derived in two different ways:

   a) If a *one-sided walls* model is requested, the final 2D model is composed by the polygons which are computed as closed rings consisting of the vectorized and completed skeleton's edges. An exemplary result can be seen in figure 4.11a).

(a)



(b)

Figure 4.11: Exemplary results: a) *One-sided walls* model overlaid on Google Maps, b) *Two-sided walls* model representing the wall widths as depicted in the photographed plan overlaid on OpenStreetMap

b) Furthermore, this method allows for the derivation of a *two-sided walls* model, i.e. a model representing the wall widths as drawn in the photographed evacuation plan. To this end, the width of each end point edge is computed as the median of the distance transformation's values extracted along the edge. This width is used to draw the symbol bridging edges into the cleaned binary image and the final polygons for the *two-sided walls* model can be detected using connected component analysis on the completed binary image. An example for a *two-sided walls* model is depicted in figure 4.11b).

4.9 Stair detection and 3D modeling

Evacuation plans are often produced by placing symbols and evacuation route information on a ground plan available e.g. as CAD model. Therefore, and because stairs are a very important part of the evacuation route information, it can commonly be observed that staircases are represented realistically in such plans. In combination with existing constraints for the height of stairs in public buildings (Neufert et al., 2002), this fact may be capitalised in order to derive an approximate floor height and extrude the 2D floor plan to a full 3D model. This involves the following steps:

1. Detection of stair candidates
   Candidates for valid stairs in public buildings generally fulfill constraints concerning their dimensions along their centre line and perpendicular to it. To be usable as stair, a minimum width (along the centre line) of 0.8 meters has to be fulfilled, while the platform's width (perpendicular to the centre line) must not fall below 0.15 meters and seldom exceeds 0.3 meters. The latter attribute is checked by computing the distance transformation for every polygon (Felzenszwalb and Huttenlocher, 2004) which enables the determination of each pixel's minimum distance to the boundary. Furthermore, the distance transformation can deliver the skeleton image which represents the centre line of the polygon. Figure 4.12 depicts an example for detected stair candidates.

2. Identification of staircases
   By searching for chains of stair candidates, outliers (like doors or windows) can be removed and staircases be identified. The search is conducted by computing the normal vector of each segment of the