Vanishing point detection for visual surveillance systems in railway platform environments

Abstract

Visual surveillance is of paramount importance in public spaces and especially in train and metro platforms which are particularly susceptible to many types of crime from petty theft to terrorist activity. Image resolution of visual surveillance systems is limited by a trade-off between several requirements such as sensor and lens cost, transmission bandwidth and storage space. When image quality cannot be improved using high-resolution sensors, high-end lenses or IR illumination, the visual surveillance system may need to increase the resolving power of the images by software to provide accurate outputs such as, in our case, vanishing points (VPs). Despite having numerous applications in camera calibration, 3D reconstruction and threat detection, a general method for VP detection has remained elusive. Rather than attempting the infeasible task of VP detection in general scenes, this paper presents a novel method that is fine-tuned to work for railway station environments and is shown to outperform the state-of-the-art for that particular case. In this paper, we propose a three-stage approach to accurately detect the main lines and vanishing points in low-resolution images acquired by visual surveillance systems in indoor and outdoor railway platform environments. First, several frames are used to increase the resolving power through a multi-frame image enhancer. Second, an adaptive edge detection is performed and a novel line clustering algorithm is then applied to determine the parameters of the lines that converge at VPs; this is based on statistics of the detected lines and heuristics about the type of scene. Finally, vanishing points are computed via a voting system to optimise detection in an attempt to omit spurious lines. The proposed approach is very robust since it is not affected by ever-changing illumination and weather conditions of the scene, and it is immune to vibrations. Accurate and reliable vanishing point detection provides very valuable information, which can be used to aid camera calibration, automatic scene understanding, scene segmentation, semantic classification or augmented reality in platform environments.

Keywords: Visual surveillance, image registration, image reconstruction, adaptive edge detection, vanishing point detection.

1. Introduction

Security often requires reliable and robust video analytics software from visual surveillance systems to monitor unstructured indoor and outdoor environments. Visual surveillance is a broad research field in computer vision that has become very active in recent years [1, 2, 3] for security and many other applications. Apart from the necessity of ensuring high levels of security in public areas and facilities, the development of computer vision devices at decreased costs as well as their miniaturization and integration in lots of environments have accelerated the use of visual surveillance systems. The increasing power of standard computing platforms, with general purpose GPU capabilities, allows computer vision tasks —implemented as software layers or modules— to be executed in real-time over the scene action provided by surveillance cameras. Video surveillance systems can be used indoors or outdoors. Applications of these systems range from security concerns, such as crime protection, prevention and forensics, to management, such as traffic and infrastructures.

Image resolution of video surveillance systems is limited by a trade-off between several system requirements, such as sensor and lens cost, transmission bandwidth and storage space, among others. However, the acquisition rate of modern video surveillance cameras makes it possible to reconstruct an enhanced image from a set of low-resolution images when the computer vision modules of the system are expected to provide accurate outputs. Multi-frame-based reconstruction techniques require a precise alignment of the set of original images to provide an enhanced image, that is, precise subpixel image registration is required.

Most man-made environments are composed of numerous buildings, roads, streets and objects that can be represented by simple volumes such as cubes or basic surfaces such as planes. These volumes and surfaces are themselves formed from elementary geometrical elements such as straight lines. These lines, when projected onto an image, intersect at vanishing points (VPs) and define the perspective of the scene. Moreover, lines are common in the type of environment being considered here (i.e. railway and underground station environments). Such environments make it easy to extract lines and VPs as they appear many times in the captured images. However, due to the large number of straight lines in scenes such as railway or underground stations, VP detection can be problematic. This is partly because of external elements such as variation in lighting, noise, distortion from the camera and occlusions. In addition, these scenes also contain a large number of people, luggage etc. which make the analysis task even more complex. This implies that any VP detection method needs to be specifically adapted to the type of scene geometry in order to offer a robust and effective system. In this paper, we aim to adapt the detection method to railway scene surveillance, an especially common security scenario, but the principles can be tuned for other environments also.

The large majority of past research into VP detection is only applied to relatively simple volumes such as cubes, parallelepipeds, or other environments such as roads or architectural images. These such environments are often termed "Manhattan" as the lines tend to be mostly parallel or perpendicular to each other. In addition, the images are generally captured with good illumination conditions and where the perspective is sufficiently extreme that the extraction of lines and VPs is relatively straightforward. The main difference between previous work and this contribution is the adaptation of well-established techniques to a different and more complex environment, where geometrical elements such as lines and VPs are so numerous that they create noise and occlusions. Furthermore, complications arise due to non-Manhattan lines such as escalator rails. For the majority of this paper, we assume Manhattan geometry, i.e. the imaged scene contains three mutually orthogonal directions. However, we also conduct preliminary experiments to show that the method has scope to locate VPs considering non-Manhattan lines near the end of the paper.

This paper presents a novel method for VP detection, improving upon our earlier works described in [4] [5], that is fine-tuned for railway and underground station environments, a common application of CCTV. Unlike much other work from the computer vision community, the proposed approach takes advantage of using real CCTV data acquired by the video surveillance system in a multi-frame image processing method, as well as the geometry of the environment in train and metro platforms. Figure 1 shows four different examples of images acquired by video surveillance systems in rail platform environments. As can be seen, the main planes and objects are delimited by straight lines denoting strong perspective effects, which is a common feature in these platforms.

The contribution of this paper is two-fold, first, we present an optimised super-resolution technique to enhance image quality in a railway platform setting. Second, we propose a novel VP detection method that is fine-tuned to railway station environments and uses the enhanced image as input. Moreover, the novelty of this work lies in the use of specific a priori knowledge and constraints related to the type of scene we are dealing with in order to compute VPs from Manhattan directions. Our algorithm is composed of four main steps:

1. Enhance the raw set of CCTV frames by noise/distortion reduction and extract lines using the Canny edge detector and the Hough transform.

2. Apply a priori information from the scene to cluster lines into those emanating from each VP.

3. Reduce the number of lines using a sub-clustering scheme before their intersections are computed based on Singular Value Decomposition (SVD).

4. Apply a voting scheme to extract the most likely VP locations from the intersections found above.

The principal lines extracted from the scene and the vanishing points computed from their intersections can be used as inputs for different modules of surveillance systems, such as automatic camera calibration [6, 7, 8, 9] and scene understanding [10, 11, 12]. Also, these inputs could be used in future developments involving robotic surveillance devices that could freely move around within the environment to perform security tasks such as surveillance, or maybe assist passengers in this type of environment. Such autonomous devices would certainly need to be able to automatically make sense of their changing 3D view as they move about. Finally, surveillance devices in platform environments could augment the view it captures to help raise alerts when anomalous behaviour is detected.

The rest of the paper is organized as follows: Section 2 examines different methods to compute an enhanced image given a sequence of images and reviews the state-of-the-art methods to detect lines and compute vanishing points from 2D images. Section 3 proposes an image enhancing technique to increase the resolving power of low-resolution images acquired by video surveillance systems in rail platform environments based on state-of-the-art methods. Section 4 describes how vanishing points are computed from images acquired in railway platforms. Finally, Sect. 5 shows the experimental results and Sect. 6 reports the main conclusions of this work.

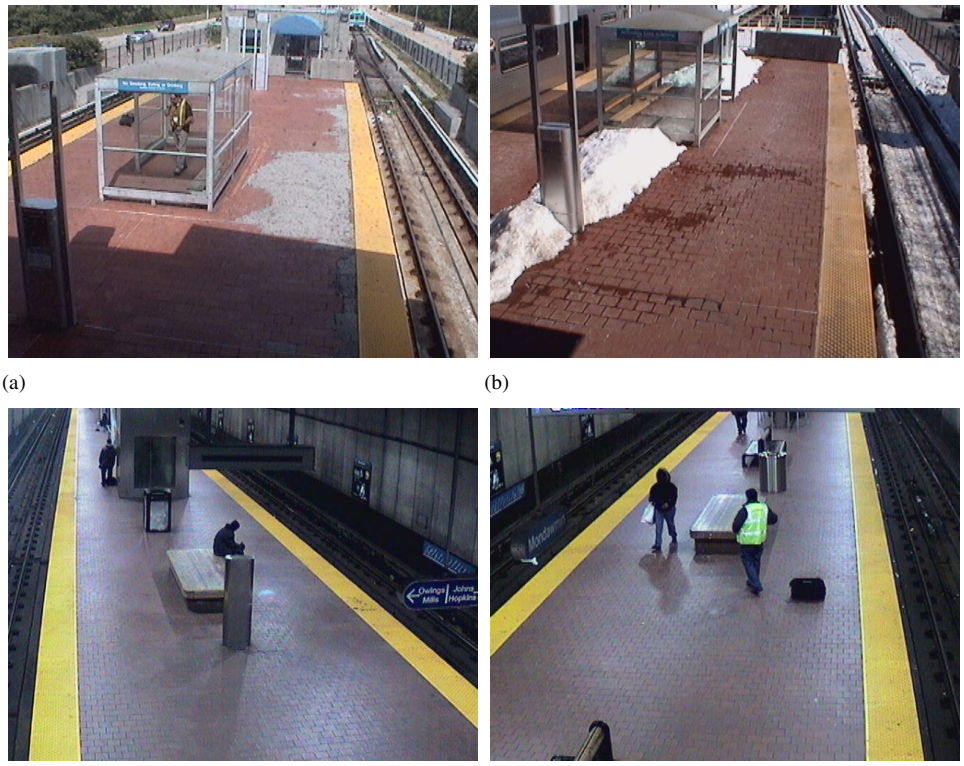
2. Related work

In this section, the previous works related with the two main topics covered in this paper are reviewed, namely: increasing the resolving power of low-resolution images and vanishing point detection.

2.1. Improving the resolving power of low-resolution images

Images provided by common visual surveillance cameras are usually of low resolution, often have poor image contrast, many times are acquired under uneven or hostile lighting conditions and, in outdoor applications, with changing weather conditions. Thus, random variations or noise appear in the images. In addition, most of the video surveillance cameras transmit the images to the host computer or the recorder system after being compressed using lossy codecs. Many visual surveillance applications can perform well with this type of images. However, if the application needs to perform accurate computations or measurements over the objects of the scene, high quality images are required. This type of application may include accurate camera calibration, accurate distance or object measurement, and accurate semantic classification of the objects of the scene. In the case of an application computing vanishing points from a scene, the higher the resolving power of the images, the higher the accuracy of the coordinates of the determined vanishing points. In these scenarios, either a hardware or a software improvement can be proposed to meet this requirement. On one hand, high-end visual surveillance cameras —featuring high-resolution sensors, high-end lenses, and even IR projectors to properly image dark scenes— can be used, though the cost of the system would be sharply increased. On the other hand, an enhanced image can be computed from the compressed, low-resolution images acquired by inexpensive cameras.

Visual surveillance cameras acquire many consecutive frames of the same scene in most of the configurations. In the case of static cameras it is always from the same orientation. But even in the case of PTZ cameras, several frames of



(a) (b) (c) (d)

Figure 1: Examples of platform environments denoting strong perspective effects: (a) and (b) outdoor rail platforms in different weather conditions; (c) and (d) indoor rail platforms.

the same orientation are often captured. Although these cameras can follow moving objects or a human operator may direct them to see another part of the scene, most of the time they remain static. Also, the acquisition rate of these cameras is high enough to get several frames from the same direction. This allows computer vision modules of video surveillance systems to use several frames to reconstruct an enhanced image reducing the signal-to-noise ratio (SNR) of the original frames and increasing the image resolving power. Furthermore, this reconstruction can also be used to remove misalignments between images acquired by surveillance cameras that can be introduced by small vibrations. In the case of rail platforms, these vibrations can be produced by trains approaching or leaving the platform, or by the wind when cameras are attached to outdoor poles.

Enhancing or improving the resolving power of images is widely studied in computer vision [13, 14]. This problem can be addressed using super-resolution (SR) methods, which aim at recovering a high-resolution image from low resolution images. These methods can be classified [15] into interpolation-based methods [16], reconstruction-based methods [17], and learning-based methods [18]. Recent SR methods are mostly learning-based, that learn a mapping between the low resolution and the high resolution image spaces [19, 20, 21, 22]. Among these methods, the Super-Resolution Convolutional Neural Network (SRCNN) has achieved superior performance than the state-of-the-art methods [23].

2.2. *Vanishing point detection*

Lines that are parallel in the real-world intersect at a common point in image space known as a vanishing point. Assuming a pinhole camera, the vanishing point of a set of lines (that are parallel in the real world) is obtained geometrically by intersecting the image plane with a ray parallel to the line of the scene and passing through the camera centre [24]. These points indicate a unique orientation in the scene which is a valuable source of information to have a better understanding of 3D geometry. The computation of VPs requires the estimation of the coordinates where lines of the image with a same real-world orientation converge. The first step is therefore the extraction of the main lines from a 2D image.

Many line detection and extraction methods have been proposed in the literature [25]. This is an active research field [26] since the approaches developed can be used in a wide range of disciplines, such as industrial [27, 28], aerial [29] and medical imaging [30] to name but a few. Among them, the technique most commonly used to detect straight lines from images is the Hough transform (HT) [31]. The Hough Transform algorithm describes features as a parameterized model used to transform them from the original image space into a 2D parameter space. A voting scheme is then used to describe how well a feature in the image fits the model. Then, a threshold is applied to the votes to effectively detect the feature. Initially intended as a

method to recover complex patterns of points in binary images [32], Hough Transform was later generalized to detect several features in images, such as lines and curves [33], and arbitrary shapes [34]. These methods are commonly referred to as standard Hough transform or non-probabilistic methods. One of the key advantages is that Hough Transform can provide very good results even in very noisy images [31]. However, standard methods require a high computation time and memory cost. Therefore, several methods, namely probabilistic methods such as the randomized Hough transform [35, 36] have been developed to reduce its complexity. Note that these methods require an edge detection of the original image prior to the application of the Hough Transform. Most commonly, the Canny edge detector is used [37].

The seminal method for detecting vanishing points is based on the projection of an image onto a Gaussian sphere centered on the optical centre of the camera [38]. The Gaussian sphere represents an accumulator space where circles correspond to lines in the image plane and their intersections to VPs. Lutton *et al.* [39] analysed different error sources in this accumulator space, such as the extension of the image, and Shufelt [40] concludes that the Gaussian sphere can lead in some occasions to spurious VPs. One alternative to the bounded accumulator space of the Gaussian sphere is the use of the image plane as an unbounded accumulator space, and the intersections of all pairs of lines in the image as accumulator cells [41].

The Hough Transform has been widely used to assist the detection of VPs in several approaches [6, 39]. Usually, VP detection methods require a high computational cost. Consequently, several approaches have been developed to tackle this issue by means of Expectation-Maximization (EM) [42], J-linkage [43], RANdom SAmple Consensus (RANSAC) [44] and M- estimator SAmple Consensus (MSAC) [45]. Conversely, other approaches are computationally more expensive since they are focused on accuracy rather than speed [41].

The use of vanishing points for camera calibration has been pioneered by Caprile and Torre [6], who proposed a method to calibrate a stereo system using simple properties of vanishing points. Wang [7] introduced the calibration of a camera using vanishing lines. More recently, Grammatikopoulos [9] proposed a method using three vanishing points of orthogonal direction and a priori information of object geometry.

Vanishing point detection is a well-studied problem that has been specialised in some scenarios to take advantage of particular features of specific environments. The most common scenario where VP detection has been specialised is in architectural environments [43, 47, 9]. In these environments high-end cameras and lenses are typically used with good illumination conditions and no vibration or other camera movement. These images are generally acquired using tripods, so no vibration effects are transmitted to the image. Furthermore, these images are processed off-line were computational complexity is not a limiting issue. Conversely, in railway platform environments, images are commonly acquired using low-resolution video surveillance cameras under uneven illumination and often with changing weather conditions. In addition, vibrations often affect the camera when trains approach or leave the platform or in the presence of strong winds. Furthermore, in outdoor platform scenes, windy weather conditions may also introduce vibrations in cameras installed in poles, as mentioned above.

3. Enhancing the resolving power of rail platform scenes

In this work, we compute an enhanced image to be used as the input for the line detection stage from a set of video frames based on two steps. These two steps are the classical steps used in super-resolution techniques [48]. First, we align the original frames using a precise sub-pixel image registration method to remove undesirable effects caused by small vibrations of the camera. Second, we reconstruct an enhanced image from the set of frames registered in the first step. After these steps the image can be corrected, if necessary, to remove the effects of the radial lens distortion. We use an enhanced image since some errors in the vanishing point detection methods come from poor line detection and extraction [39, 47], which introduce errors in the coordinates of the main lines of the scene, and thus, in the coordinates of the computed vanishing points. Figure 2 shows the architecture of the proposed approach.

3.1. Image registration

Image registration can either be done in the spatial or frequency domain. On one hand, methods based on the spatial domain can deal with general motion models, such as homographies. On the other hand, methods based on the frequency domain are restricted to global motion models and usually only consider planar shifts, planar rotations and scale, which can be easily managed in the Fourier domain. Most of the state-of-the-art image registration methods have difficulties when dealing with noisy images. This is the case of the images acquired by video surveillance systems in rail platform environments due to the use of low-cost cameras and/or the use of lossy compression methods to transmit the acquired frames.

In this work, we align the low resolution frames, I_{LR}, acquired by video surveillance cameras in train and metro platforms using a procedure built on the frequency-based method to estimate image rotations proposed by Vandewalle *et al.* [49]. This method outperforms other state-of-the-art frequency domain methods and also performs better than spatial domain methods if the image presents some directionality, as it is the case of railway platform images showing strong perspective effects (see Figure 1). This process follows the next steps:

1. Multiply the frames $I_{LR,n}, n \in [1, N]$, by a Tukey window to make them circularly symmetric.
2. Compute the Fourier transforms $F_{LR,n}$ of all low-resolution images.
3. Estimate the rotation angles, ϕ_n, between every frame $I_{LR,n}, n \in [2, N]$ and the reference frame $I_{LR,1}$.
4. Estimate the vertical and horizontal translations, Δx_n, between every frame $I_{LR,n}, n \in [2, N]$ and the reference frame $I_{LR,1}$.

In this alignment process, only low-frequency information of the images is used, that is, the part of the image with the

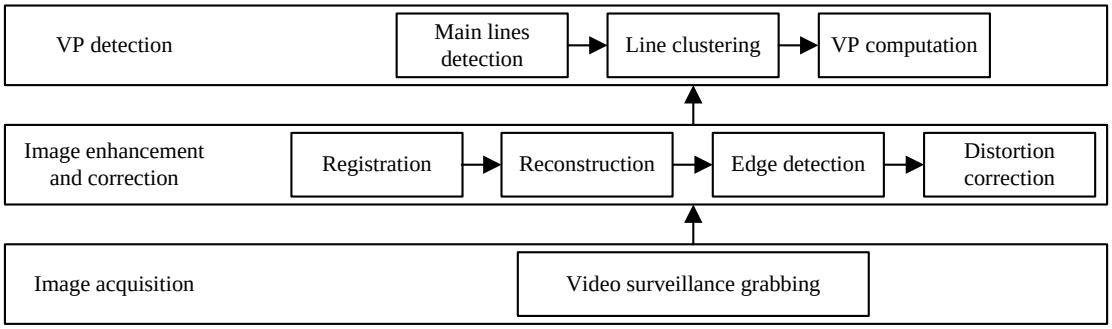


Figure 2: Architecture proposed for VP detection in outdoor and indoor rail platform environments.

highest SNR. This part of the image is aliasing-free and, thus, if the surveillance camera (or the codec) provides aliased images, this method can deal with this issue. In frames acquired from railway platform environments aliasing may appear in long distances when imaging rail ties or ground tiles, for instance. Two requirements are imposed to properly align the original images: the original images are undersampled, and the imaged scenes cover large distances. These requirements are met by the images acquired in railway platform scenes.

3.2. Image reconstruction

Image reconstruction can be achieved by means of interpolation methods, statistical techniques and Bayesian estimation [50, 51], iterative backprojecton and the projection onto convex sets (POCS) algorithm [52] among others. For our image enhancer, we chose interpolation due to its ideal balance between performance and computational cost. Once the original frames are accurately aligned, an enhanced image can be computed. First, an estimation of the optimal number of frames, n_f, to be used in the reconstruction step is required.

The video sequences used in all the experiments shown in this paper were acquired using Sony SNC-DF70 surveillance cameras, equipped with a $1/4''$ Sony Super HAD (Hole Accumulation Diode) CCD sensor with an effective resolution of 768 x 494 pixels, capable of providing images up to 640 x 480 pixels. We ran an experiment over six video sequences acquired from three outdoor and three indoor railway platform environments to determine the optimal number of frames to use in the reconstruction step. This experiment was carried out over the frames provided by the video surveillance cameras compressed using MPEG-4 (image resolution was set to 640 x 480 pixels). One frame from each video was randomly selected, acting as ground truth. Starting with the frame after the selected one in each video sequence, a low-resolution image (with a 320 x 240 pixel resolution) was computed from each frame of the experiment by subsampling. Then, several images were reconstructed (with a 640 x 480 pixel resolution) using a different number of frames. Figure 3 shows the mean-squared error (MSE) of the reconstructed image as a function of the number of the original frames used. The reconstruction step provides the best relationship between performance and computational cost when five to seven input frames are used. Thus, in this work we determine

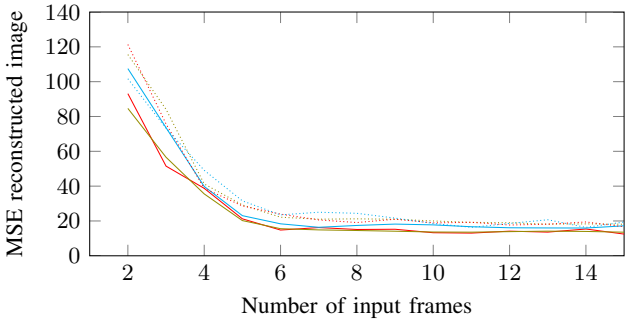


Figure 3: MSE of the reconstructed image as a function of the number of input frames; the solid line represents the outdoor scene and the dotted line the indoor scene.

$n_f = 6$, that is, six frames are used in the reconstruction of the enhanced image for further stages.

Once the number of aligned frames to be used in the reconstruction step is determined, the optimal time interval between the acquisition of these frames, t_f (in seconds), must be estimated. We ran an experiment over six different video sequences acquired by surveillance cameras in rail platforms to determine the best gap between frames to reconstruct an enhanced image. The video sequences used in this experiment were acquired when no train was approaching or leaving the platform and with no people in the platform. Figure 4(a) shows an example frame of each of the video sequences used in this experiment. A ground truth image was randomly chosen from each video sequence. Then, sets of n_f frames were extracted from each video, right after the frame considered as the ground truth image, using different time intervals between the frames. After that, a low-resolution image (with a 320 x 240 pixel resolution) was computed from each frame. Finally, an enhanced image from each set of frames was reconstructed. Figure 4(b) shows the MSE of the reconstructed image as a function of the time interval. As can be seen, the best results are obtained when the input frames are chosen with a gap between two and three seconds, $t_f \in [2, 3]$. Shorter gaps do not allow to enhance the quality of the image. Larger gaps introduce more noise maybe due to changes in illumination (this can be seen because the MSE in the reconstructed images from the indoor mainly re-

mains constant regardless the gap used).

The reconstruction of the high-resolution image follows the next steps:

1. For every image $I_{LR,n}, n \in [2, n_p]$, compute the coordinates of its pixels in $I_{LR,1}$, using the registration parameters, ϕ_n and Δx_n, estimated for each frame.
2. Interpolate the values on a high-resolution grid using cubic interpolation.

The simplest method to fuse several images is averaging. Using this method, the resolving power of the enhanced image is not increased compared to the original ones. We also ran an experiment averaging all the original frames to obtain an enhanced image. Although this method is only capable of reducing noise and does not increase the resolving power of the image, it can be used to estimate the error obtained in the reconstruction step using the frequency-based method. Figure 4(c) shows the MSE of the images enhanced by averaging the original frames as a function of the time interval. As can be seen, the frequency-based reconstruction method may increase or even reduce the level of noise compared to the average-based method. In the worst case, the error is double that of the error introduced by the averaging method. However, in the frequency-based reconstruction method the size of the enhanced image is quadrupled and, most importantly, the resolving power is increased.

Figure 5 shows the effect of edge detection from an image enhanced using the procedure described above. Figure 5(a) shows the original frame of a video sequence acquired in an outdoor railway platform; Figure 5(b) and Figure 5(c) show the output of an edge detection performed over the original frame and over the enhanced image in a given region, respectively. The edge detection was carried out using the Canny edge detector with equivalent hysteresis thresholds and values for smoothing, taking the difference between the size of the images into account. As can be seen, there are some hints on the edge detection that reveals that the resolving power of the enhanced image has been improved. First, in the original frame only, two edges of the railway tracks were detected, whereas three edges were detected in the enjavascript:void(0);hanced image. An accurate detection of the railway tracks is of paramount importance when computing the vanishing points of the scene, since they are known to be parallel lines in the real world and their intersection in the coordinate system of the image is a clear vanishing point. Second, the edges of the vertical structure on the left of the shelter are better identified in the enhanced image. This information can be of utmost importance for a semantic segmentation of the objects of the scene. Finally, the yellow area on the right edge of the platform is also better identified in the enhanced image.

Although the frequency-based reconstruction method built based on [49] is computationally efficient, we developed a pipelined architecture which allows performing the image enhancer online. In this architecture, all the stages can run in parallel and once a new image is available for the enhancing stage, it is added to the previous images that have already been registered and the enhanced image is reconstructed very fast. Figure 6 shows a diagram of the pipeline evolution for a sequence of input frames I_i, $i \in [1, 15]$, where $t_0, \ldots t_{15}$ are separated by the interval specified in t_f, I_{i-j} is the image registered using input frames I_i to I_j, I'_{i-j} is the image reconstructed using registered image I_{i-j}, E_{i-j} is the edge detection over image I'_{i-j}, UI'_{i-j} is the undistorted version of I'_{i-j} computed using E_{i-j}, and VP_{i-j} is the vanishing point output computed as a result of input frames I_i to I_j.

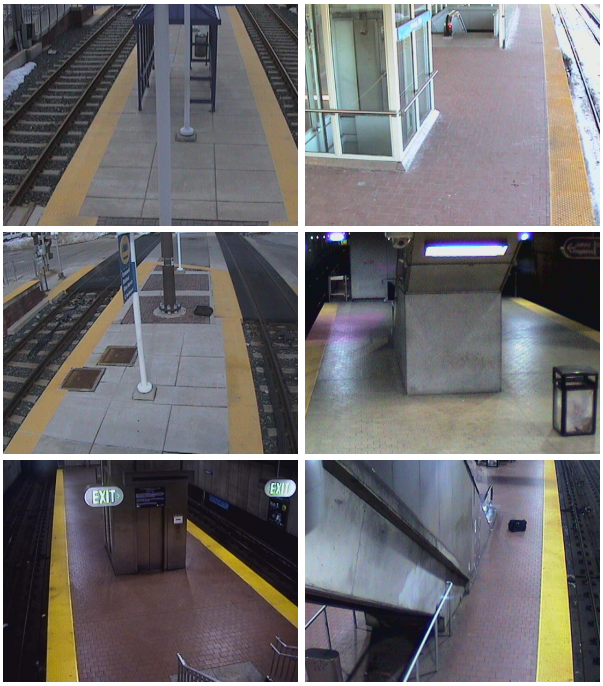
As mentioned above, after the image has been reconstructed, a correction step could be added to remove the radial lens distortion to the enhanced image. The effects of the radial lens distortion in the image can be removed without user intervention using the main straight lines of the image —detected in the next stage. Since the vanishing point computation involves extracting the main straight lines from the image, the radial lens distortion of the camera can be determined based on a least-squares adjustment of the points belonging to these lines, as proposed in [9]. Figure 7 shows an example of an image enhanced using the procedure described above. Once the image was enhanced, its radial lens distortion was corrected. In this example, a moving person appears in some of the input frames of the image enhancer (see Figure 7(a)) but the registration step diffused this area. Thus, it does not affect further stages, and enables data to be captured in the presence of passengers. Figure 7(b) shows the enhanced and corrected images.

4. Vanishing point detection in rail platform scenes

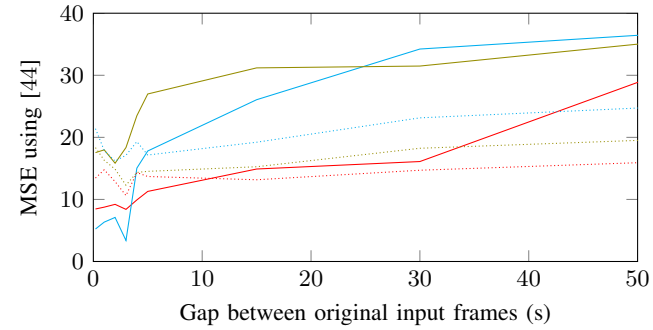
Images from video surveillance systems are mostly of low resolution, which makes scene analysis a difficult task. However, the method presented in the previous section allowed to increase the resolving power of the image and thus to get better results for line extraction and clustering and VP computation stages. Images with a high resolving power in video surveillance systems can be used to accurately perform several computer vision operations. In this work, the images enhanced as described in the previous section constitute the input of a vanishing point detection procedure (see Figure 2). The procedure described in this work follows a three-step approach for vanishing point detection from 2D images: first, the main lines of the image are detected, second, the lines are clustered according to a priori knowledge from the scene, finally the coordinates of the vanishing points are computed. We compute all possible vanishing points from the image. Then, geometry restrictions are applied over the set of provided vanishing points to compute those corresponding to Manhattan directions.

4.1. Line detection

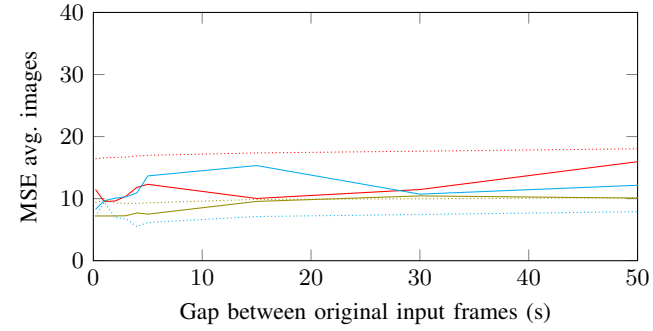
In this section, we present the method for extraction and clustering of lines before computation of the VP coordinates. Our input is a greyscale frame F captured by the CCTV camera and enhanced using the methods mentioned in the previous section. Note that the above methods typically use multiple input frames to obtain F and are therefore somewhat robust to the movement of pedestrians etc. which will effectively be averaged out across frames. The Canny edge detector is then applied to extract



(a)

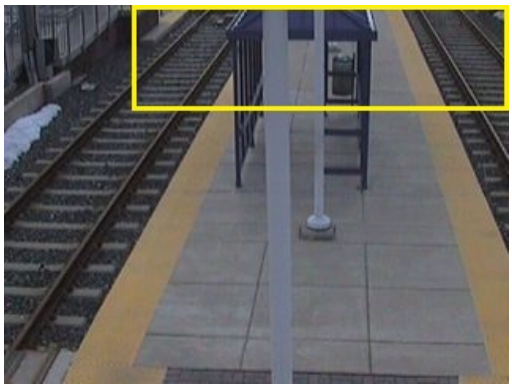


(b)

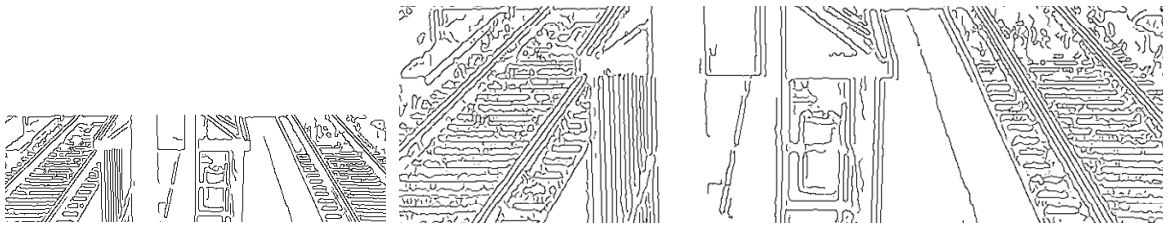


(c)

Figure 4: Effects of the multi-frame image enhancing: (a) examples of frames of the video sequences used in the experiment; (b) MSE of the enhanced image using frequency-based reconstruction as a function of the gap used to select the frames from the video sequence; (c) MSE of the enhanced image using averaging as a function of the gap used to select the frames from the video sequence; in (b) and (c) solid lines represent outdoor scenes and dotted lines represent indoor scenes.



(a)

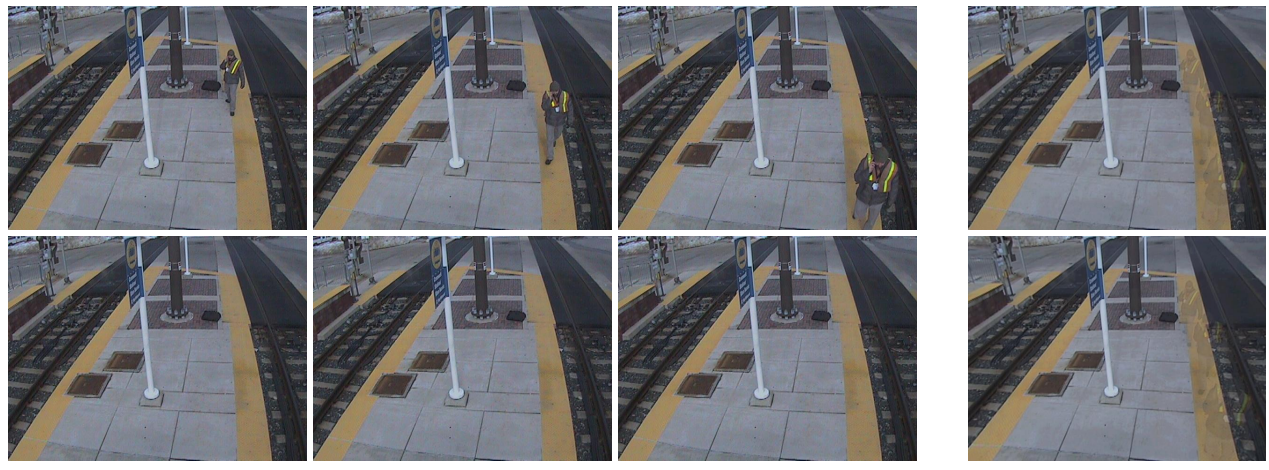


(b) (c)

Figure 5: Edge detection from an original, low resolution frame and from and enhanced, high resolution image obtained using the multi-frame image enhancer described in this paper (the resolution of the latter is double the resolution of the former): (a) original low-resolution frame (yellow lines highlight a given region of interest; (b) edge detection from the original frame in the given region; (c) edge detection from the enhanced image in the given region.

Stage \ time	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}	t_{12}	t_{13}	t_{14}	t_{15}
Image acquisition	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}	I_{11}	I_{12}	I_{13}	I_{14}	I_{15}	
Image registration							$I_{1\text{-}6}$	$I_{2\text{-}7}$	$I_{3\text{-}8}$	$I_{4\text{-}9}$	$I_{5\text{-}10}$	$I_{6\text{-}11}$	$I_{7\text{-}12}$	$I_{8\text{-}13}$	$I_{9\text{-}14}$	
Image reconstruction								$I'_{1\text{-}6}$	$I'_{2\text{-}7}$	$I'_{3\text{-}8}$	$I'_{4\text{-}9}$	$I'_{5\text{-}10}$	$I'_{6\text{-}11}$	$I'_{7\text{-}12}$	$I'_{8\text{-}13}$	
Edge detection									$E_{1\text{-}6}$	$E_{2\text{-}7}$	$E_{3\text{-}8}$	$E_{4\text{-}9}$	$E_{5\text{-}10}$	$E_{6\text{-}11}$	$E_{7\text{-}12}$	
Distortion correction										$UI'_{1\text{-}6}$	$UI'_{2\text{-}7}$	$UI'_{3\text{-}8}$	$UI'_{4\text{-}9}$	$UI'_{5\text{-}10}$	$UI'_{6\text{-}11}$	
VP detection											$VP_{1\text{-}6}$	$VP_{2\text{-}7}$	$VP_{3\text{-}8}$	$VP_{4\text{-}9}$	$VP_{5\text{-}10}$	

Figure 6: Pipeline evolution.



(a) (b)

Figure 7: Example of an image showing a platform environment enhanced and corrected using the procedure described in this paper: (a) original frames acquired with a two-second time interval between them (first frame of the sequence is top left; last frame is bottom right); (b) enhanced image after registration and reconstruction with more resolving power (above) and enhanced image after correcting the radial lens distortion (below).

edges from the enhanced image. It is well known that three parameters must be set for the Canny edge detector: the size of the Gaussian filter used to smooth the image in the first step of the detector, σ, and the hysteresis thresholds used to extract the edges, t_l and t_h. A common size of the smoothing kernel can be experimentally determined for both outdoor and indoor scenes of railway platform environments. However, due to changes in illumination conditions, it is not possible to get a common value for the hysteresis thresholds without missing lines or adding noise to the final result when these conditions change. Thus, an auto-adaptive method to compute these thresholds should be used.

Although there is no optimal solution —because there is not a unique edge segmentation of the image; it depends on what this segmentation is going to be used for in high-level computer vision modules— several auto-adaptive methods have been proposed to deal with this task. Most of them are based on the Otsu method [53], which relies on the gradient magnitude of the image and searches for the threshold, t_O, that minimizes the intra-class variance, or equivalently, ensures that the inter-class variance is maximal. In this method, an image is defined as $I(x, y)$, and L being the number of distinct gray levels, n_i being the number of pixels with gray level i, and N the number of pixels of the image. In this image, the probability of a pixel having the gray level i is defined in Eq.(1).

$$p_i = \frac{n_i}{N} \tag{1}$$

If the image is divided into two classes, C_0 and C_1, by a threshold in level k, C_0 denotes pixels with levels $[0, k]$ and C_1 pixels with levels $[k + 1, L]$. The optimal threshold, t_O, the cumulative probabilities $P_0(k)$ and $P_1(k)$ can be obtained using Eq.(2) and Eq.(3), respectively. Also, the mean levels of C_0 and C_1, μ_0 and μ_1 can be computed using Eq.(4) and Eq.(5), respectively.

$$P_0(k) = \sum_{i=0}^{k} p_i \tag{2}$$

$$P_1(k) = \sum_{i=k+1}^{L} p_i \tag{3}$$

$$\mu_0(k) = \sum_{i=0}^{k} i \frac{p_i}{P_0(k)} \tag{4}$$

$$\mu_1(k) = \sum_{i=k+1}^{L} i \frac{p_i}{P_1(k)} \tag{5}$$

Then the optimal threshold t_O can be obtained as Eq.(6).

$$t_O = \arg \max_{1<k<L} \left((P_0(k)(\mu_0(k))^2 + (P_1(k)(\mu_1(k))^2 \right) \tag{6}$$

In some circumstances, for example when the histogram distribution of the image is unimodal, the threshold provided by the Otsu method will be incorrect. Thus, some improvements of the Otsu method have been proposed [54, 55, 56]. Many image processing applications use t_O as the high threshold for the Canny edge detector. Because of the nature of the scenes, unstructured indoor and outdoor environments will provide bimodal or multimodal distributions. Thus, the threshold provided by the Otsu method will be used as t_h in the proposed approach. Then, we set the low hysteresis threshold as $t_l = 0.5 \cdot t_h$ (following the criterion of the high-to-low ratio between 2:1 and 3:1 recommended by Canny [37]).

Once edges have been extracted, the main lines of the image are detected by means of the standard Hough Transform. In this stage, we promote only long lines of the enhanced image to pass to the vanishing point detection stage. Thus, a threshold, l_t, is established for the voting scheme in the Hough Transform to discard short segments. As mentioned above, the input of the Hough Transform algorithm is a description of the edges of the image. To set the length threshold used by the Hough Transform we ran an edge and further line detection experiment over different rail platform scenes with several values for l_t. We conclude $l_t = 100$ provides a good estimator for the length and high hysteresis threshold required for an accurate line detection for image resolution values used in this paper.

4.2. *Vanishing point computation*

We compute vanishing points in the images acquired from railway platform scenes using the image plane as the accumulator space, as in [41] and in [9]. We chose this unbounded accumulator space since it preserves all geometrical information from the original image and it does not require camera calibration as opposed to the Gaussian sphere. Each main line of the image is assigned to a vanishing point, or identified as an outlier.

We introduce knowledge about the railway platform scenes to reduce the computational complexity and unsuccessful detection of vanishing points.

First, lines are clustered taking into account the dominant directions according to their angle, as in [57]. In general, in railway platform environments two main directions lie close to the vertical and the horizontal direction. Second, most video surveillance cameras in railway platforms are installed in a high position and the z axis of the scene follows the direction of the rail tracks; thus, a vanishing point is expected above the image boundary —where lines from rail tracks and platform edges in the 3D scene converge in the 2D image plane— (see Figure 1, Figure 4(a) and Figure 5). Third, most of the time all vanishing points corresponding to Manhattan directions in the scene are located beyond the image boundaries. If a vanishing point is located inside the image boundaries it probably comes from a plane which is non-orthogonal with the Manhattan directions of the scene. These points in rail platform scenes usually come from stairways or escalators (see the image on the right of third row in Figure 4(a)).

We compute all pairwise intersections of lines pointing to a similar direction (each cluster) and mark each intersection as a putative vanishing point. Since we only promote long lines in the Hough Transform-based line detection stage, we consider that all these lines can point toward a possible vanishing point.

Then, we compute the vanishing point as the centroid of all intersections by least squares using singular value decomposition (SVD). We use a combinational approach to compute all the pairwise line intersections since in rail platform environments the number of principal lines is not very large and do do not incur in a big performance penalty.

The most expensive task is computing the line intersections, but this operation can be done using SVD on a GPU, so it is highly parallelizable. If the combinational approach would not meet the deadlines imposed by the high-level computer vision modules of the surveillance system, line clustering could be carried out using RANSAC, as in [58], and some intersections would not be computed since minimal sets of lines could be used to reach a consensus.

Although computing the vanishing point using the centroid of all the line intersections does not provide the optimal solution [24], its accuracy is high enough for most of the applications when the edges of the image have been accurately extracted. Also, it can be computed very fast. However, if the accuracy obtained using this method would not meet the requirements of the high-level computer vision modules, it could be refined using EM, or another iterative algorithm, or computing the maximum likelihood estimate (MLE) of the intersections.

Finally, when two lines in the same cluster, $l_1\,(\rho_1, \theta_1)$ and $l_2\,(\rho_2, \theta_2)$ (ρ_1 and ρ_2 are the distance from the origin to the closest point on the straight line 1 and 2 respectively and θ_1 and θ_2 are the angle between the x axis and the line connecting the origin with that closest point), have the same angle, $\theta_1{=}\theta_2$, the vanishing point is at infinity —that cannot be represented in the image coordinate system. Thus, we compute this vanishing point as the farthest point in the accumulator space, Δ_{max}, with parameters $\theta = \theta_1 = \theta_2$ and $\rho = \frac{1}{2}(\rho_1 + \rho_2)$, as proposed in [47].

5. Experimental results

This section presents a number of experimental results to demonstrate the effectiveness of the proposed approach to robustly detect vanishing points from low-resolution images acquired by video surveillance systems in rail platform environments. Firstly, the most representative video sequences acquired from video surveillance systems are selected, both from outdoor and indoor platform environments. The aim of this selection is to look for different orientations of the rail tracks in the scene, different locations of platform furniture, such as shelters and benches, and different illumination conditions. Secondly, the ground truth of several frames from each video sequence is computed manually. And, finally, the vanishing points of each frame are automatically computed using two approaches: that proposed in this paper and another method to compare the obtained results to.

To the best of our knowledge there are no specific approaches intended to compute vanishing points from rail platform environments —as reviewed above, most of the related work is focused on architectural environments. Thus, we choose the approach proposed by Tardif [43] which provides very good results in this type of environment and is based on a non-iterative method using J-linkage to cluster the lines of the image according to the vanishing point they pertain to. Although the method proposed by Tardif does not enforce orthogonality of the vanishing points when generating the hypotheses and generally, its only drawback compared to more recent methods is that it requires a larger number of line segments to converge to the correct solution [59].

In these experiments, 18 video sequences acquired from 18 different rail platform environments were used; 10 from outdoor and 8 from indoor platforms. Five starting frames were equidistantly selected from each clip to ensure that the experiments are run over images with different lighting conditions. Therefore, the dataset for the experiments consisted of 90 frames (40 indoor and 50 outdoor).

The ground truth for these experiments consisted of the vanishing points of each frame. The vanishing points were labeled by people who did not develop the approach proposed in this paper to avoid any bias due to prior knowledge of the vanishing point computation. These people were trained on the usage of a very basic piece of software to annotate in the image two main lines pointing to each main direction of the scene, that is, to identify two main lines that converge to each vanishing point. Then, this software computed automatically the intersections of these lines and labeled them as the ground truth for the image. Figure 8 shows two examples of the manual annotation.

Once the ground truth had been determined, we computed the vanishing points in the dataset using two automated approaches. Firstly we ran the three-stage approach proposed in this paper. An enhanced image is computed using several frames (n_f), as described in Sect. 3, and then, the vanishing points from the enhanced image are computed as described in Sect. 4. Secondly, we ran the approach proposed by Tardif, using the implementation[1] provided by the author in [43], with the same input frames.

One of the key points of these experiments is classifying a candidate vanishing point as a real vanishing point and comparing to the ground truth. Several approaches can be used. Schmitt and Priese [47] propose a method based on a reference point in the image (e.g. a known feature or object of the scene) and the computation of the angle formed by the two lines connecting this reference point. If this angle is below a given threshold the vanishing point is successfully determined; otherwise, it is a miss. In this work we use the Euclidean distance to determine when a vanishing point is successfully or unsuccessfully detected. A fixed threshold to determine whether the candidate point is correct is inappropriate because VPs far from the image centre (especially when outside the image) do not need to be as close to the ground truth to be deemed successfully detected. For this reason, we use a relative threshold for each candidate vanishing point. We compute the Euclidean distance from the image center to the candidate vanishing point, d_{oc}, and we consider the vanishing point successfully determined if the Euclidean distance between the candidate vanishing point and the ground truth, d_{cg}, is shorter than the threshold $t_d = 0.02\,d_{oc}$, that is, it is less than 2% of d_{oc}.

[1] https://github.com/borist/cis400/tree/master/VPdetectionTardif

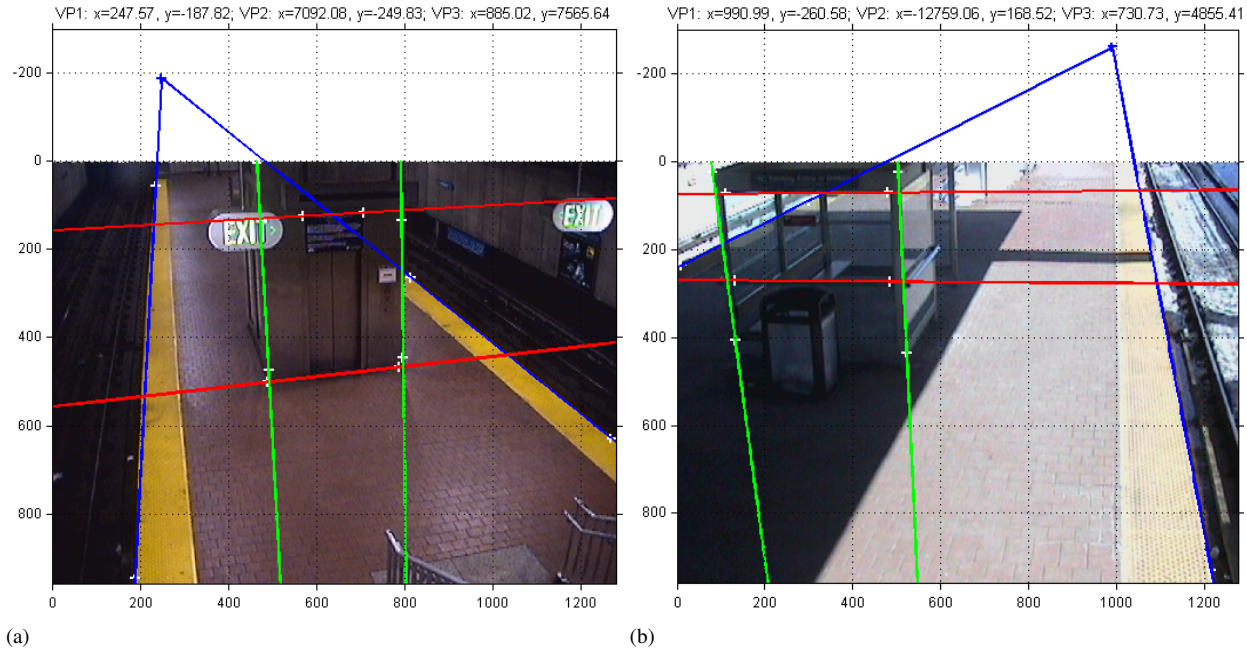


Figure 8: Example of manual annotation of two main lines pointing to each main direction of the scene to further compute the ground truth of the frame: (a) indoor platform scene; (b) outdoor platform scene (due to space limitation, only the vanishing point in the direction of the rail tracks is shown though all the lines annotated and the coordinates of the three vanishing points are displayed).

The values for the parameters required by the approach proposed in this work are determined empirically following a factorial design experiment. The values used in the experiments carried out in this work are shown in Table 1.

Using the aforementioned definition to classify candidate vanishing points, the experimental results are shown in Table 2. As can be seen, 11 out of a total of 18 vanishing points were detected, resulting in a 61.1% hit rate (increased to 78.0% for $t_d = 0.05$). Conversely, the approach proposed by Tardif [43] provided 5 vanishing points, a 27.8% hit rate (increased to 44.0% for $t_d = 0.05$). In our method, 6 out of 10 vanishing points were detected in outdoor platform environments (60.0%), whereas 5 out of 8 were detected in indoor scenes (62.5%). The approach proposed by Tardif could detect 2 and 3 (20.0% and 37.5%), respectively. Table 2 also shows results for the proposed method without the image enhancement method: although all VPs were detected, none of them were within the required threshold. Finally, Table 3 shows the mean errors for the various successfully and unsuccessfully detected VPs indicating notable improvement over the previous research, provided that the image enhancement is completed first. The mean error represents the percentage error calculated as follow. For example, if the detected VP is 10 pixels away from the ground truth value, and the ground truth is 200 pixels from the image centre, then the percentage error would be 10/200 = 5%. We then have a percentage error for each image sequence and we take the mean of these for each row of table 2.

The key factor for these differences is twofold. On one hand, the auto-adaptive edge and line detection carried out in the approach proposed in this work provides a more accurate edge map and more main lines can be accurately detected. In the method proposed by Tardif some of the unsuccessfully detected or the undetected vanishing points came from frames where a reasonable number of straight lines has not been generated in the line detection stage. This also happens in other methods, such as that proposed by Schmitt and Priese [47], that do not take into account the ever-changing illumination conditions of the image. Since we propose the use of an adaptive edge detection as a base to detect the main lines, these conditions are taken into account, and thus, more vanishing points can be successfully detected. On the other hand, our approach avoids very short segments being passed to the vanishing point detection stage, while long lines are promoted. Some of the unsuccessfully detected vanishing points in the approach of Tardif came from short segments detected in the image. As mentioned above, we solve this issue applying specific knowledge about railway platform scenes. Moreover, for future work, the lines could be weighted by their length or length squared.

A qualitative analysis of results shows relatively strong performance against weather and time of day but poorer performance in the presence of non-Manhattan lines or were no strong linear features are present in the image. Figure 9 shows a typical example of a failure case, which suffers from both of these complications. Such cases are especially challenging to our method and are yet, unfortunately, relatively common in railway stations. Potential methods to overcome such limitations in future work include combining information from multiple viewpoints in a network of cameras and using the dynamical information from passenger movement to differentiate inclined planes from Manhattan features.

Parameter	Value	Description
n_f	6	Number of frames used in the multi-frame enhancing stage
t_f	2	Time interval between input frames for the multi-frame enhancing stage (in seconds)
σ	1.5	Size of the Gaussian filter used for image smoothing
l_t	100	Minimum length of the lines promoted in the Hough Transform-based line detection (in pixels)
t_d	0.02	Distance threshold to determine a VP from a candidate point

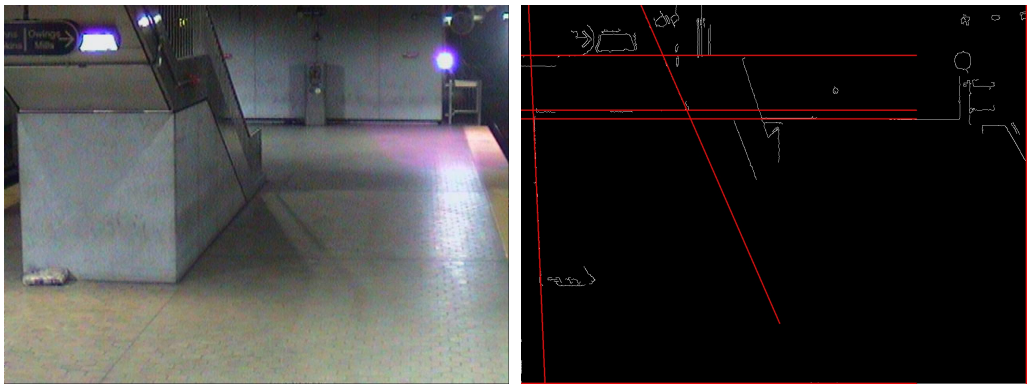
Table 1: Values given to the parameters required by the approach proposed in this work for the experiments carried out in this work.

	Ground truth	Proposed approach (LR)	Proposed approach (HR)	Tardif [43]
Vanishing points successfully detected	18	0	11 (61.1%)	5 (27.8%)
Outdoor scenes	10	0	6 (60.0%)	2 (20.0%)
Indoor scenes	8	0	5 (62.5%)	3 (37.5%)
Vanishing points unsuccessfully detected	–	18 (100%)	5 (27.8%)	10 (55.6%)
Outdoor scenes	–	10 (100%)	4 (40%)	8 (80.0%)
Indoor scenes	–	8 (100%)	1 (12.5%)	2 (25.0%)
Undetected vanishing points	–	0	2 (11.1%)	3 (16.7%)
Outdoor scenes	–	0	0	0
Indoor scenes	–	0	2 (25.0%)	3 (37.5%)

Table 2: Experimental results obtained from 18 different rail platform scenes for the proposed method using high-resolution images (HR – from super-resolution), low resolution images (LR – using raw data only) and the method of Tardif.

	Mean error (LR)	Mean error (HR)	Mean error Tardif [43]
Vanishing points successfully detected	–	1.25	1.25
Outdoor scenes	–	1.13	1.29
Indoor scenes	–	1.37	1.20
Vanishing points unsuccessfully detected	93.13	5.08	20.87
Outdoor scenes	126.28	7.81	34.75
Indoor scenes	59.97	2.34	6.99

Table 3: Mean error obtained from 18 different rail platform scenes.



(a) (b)

Figure 9: Example case with no VPs detected.

6. Conclusions

In this work, we propose a robust approach to accurately detect the main lines and vanishing points in low-resolution images acquired by video surveillance systems in indoor and outdoor rail platform environments. The proposed approach consists of three stages. Firstly, several frames acquired by the video surveillance system are used to increase the resolving power of the image through a multi-frame image enhancer. The sequence of frames are registered using a frequency domain approach and an enhanced image is reconstructed by interpolating the aligned, low-resolution images. Secondly, an adaptive edge detection is performed, distortions of the image are corrected and the main lines of the image are extracted. Finally, vanishing points are detected taking into account specific knowledge about rail platform scenes.

The experiments carried out in this work demonstrate the effectiveness of the proposed approach. The results were compared with a method which provides very good results for detecting vanishing points in Manhattan-like scenes. However, that method (and related work), is specifically designed for architectural environments, where high-end cameras equipped with high-resolution sensors are typically used and images can be acquired with good illumination conditions. By contrast, this paper focused on a different environment where we applied specific knowledge of railway station environments. To the best of our knowledge, this is the first method to specialise on this type of scene. While we do not claim that the proposed method is superior to all other methods for generic scenes, the results show that, for our target application, the method is better suited than the existing state of the art.

The proposed technique will reduce time and cost of commissioning automated scene analysis systems that use fixed cameras as it allows automatically detecting VPs without any user intervention and can be used to calibrate video surveillance cameras if metric measurements of the scene were required. It can also be applied to moving cameras to determine their new position and assist with automatic calibration.

Finally, it can also be used to aid high-level computer vision modules of video surveillance systems to segment the scene — including partially obscured objects— and perform a semantic classification of the objects and main planes in railway platforms, and thus, to identify locations that are or are not allowed for people to walk or stand, for instance. These locations could be notified to human security operators using augmented reality based on the provided information. In addition, tracking the position of the vanishing points can be used to determine the movement of moving cameras (PTZ) or future robotic surveillance devices that could freely move around a platform environment.

Acknowledgements

This work was funded by HEFCE quality-related research (QR) and Aralia Systems Ltd. under project P44G1076, and by the Ministry of Education of Spain under the National Program for Mobility of Human Resources of the 2008-2011 National Plan of Research, Development and Innovation. The authors would like to thank the technicians and engineers of Aralia Systems Ltd. for providing the images and video sequences of railway stations used in this work.

References

1. Haering, N., Venetianer, P.L., Lipton, A.. The evolution of video surveillance: An overview. *Machine Vision and Applications* 2008;19(5-6):279–290.
2. Remagnino, P., Velastin, S.A., Foresti, G.L., Trivedi, M.. Novel concepts and challenges for the next generation of video surveillance systems. *Machine Vision and Applications* 2007;18(3-4):135–137.
3. Hu, W., Tan, T., Wang, L., Maybank, S.. A survey on visual surveillance of object motion and behaviors. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews* 2004;34(3):334–352.
4. Tarrit, K., Atkinson, G., Smith, M., Molleda, J., Wright, G., Gaal, P.. Vanishing point detection for cctv in railway stations. In: *5th International Conference on Imaging for Crime Detection and Prevention*. IET. ISBN 978-1-84919-904-9; 2013:.
5. Tarrit, K.. An investigation into common challenges of 3D scene understanding in visual surveillance. *PhD Thesis*. 2017:.
6. Caprile, B., Torre, V.. Using vanishing points for camera calibration. *International Journal of Computer Vision* 1990;4(2):127–139.
7. Wang, L.L., Tsai, W.H.. Camera calibration by vanishing lines for 3-D computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1991;13(4):370–376.
8. Kosecka, J., Zhang, W.. Video compass. *Lecture Notes in Computer Science* 2002;2353:476–490.
9. Grammatikopoulos, L., Karras, G., Petsa, E.. An automatic approach for camera calibration from vanishing points. *ISPRS Journal of Photogrammetry and Remote Sensing* 2007;62(1):64–76.
10. Barinova, O., Konushin, V., Yakubenko, A., Lee, K., Lim, H., Konushin, A.. Fast automatic single-view 3-d reconstruction of urban scenes. *Lecture Notes in Computer Science* 2008;5303:110–113.
11. Wan, G., Li, S.. Automatic facades segmentation using detected lines and vanishing points. In: *Proceedings - 4th International Congress on Image and Signal Processing, CISP 2011*; vol. 3. 2011:1214–1217.
12. Hernndez, D.C., Jo, K.. Stairway segmentation using Gabor filter and vanishing point. In: *2011 IEEE International Conference on Mechatronics and Automation, ICMA 2011*. 2011:1027–1032.
13. Pitas, I.. Digital Image Processing Algorithms and Applications. Wiley-Blackwell; 2000.
14. Szeliski, R.. Computer Vision: Algorithms and Applications. Springer-Verlag New York Inc; 2010.
15. Timofte, R., Rothe, R., Van Gool, L.. Seven ways to improve example-based single image super resolution. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*; vol. 2016-January. 2016:1865–1873.
16. Thvenaz, P., Blu, T., Unser, M.. Image interpolation and resampling. Handbook of Medical Image Processing and Analysis; 2009:465–493.
17. Yang, J., Wright, J., Huang, T.S., Ma, Y.. Image super-resolution via sparse representation. *IEEE Transactions on Image Processing* 2010;19(11):2861–2873.
18. Zhang, K., Tao, D., Gao, X., Li, X., Xiong, Z.. Learning multiple linear mappings for efficient single image super-resolution. *IEEE Transactions on Processing* 2015;24(3):846–861.
19. Yang, C., Yang, M.. Fast direct super-resolution by simple functions. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2013:561–568.
20. Timofte, R., De, V., Gool, L.V.. Anchored neighborhood regression for fast example-based super-resolution. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2013:1920–1927.
21. Cui, Z., Chang, H., Shan, S., Zhong, B., Chen, X.. Deep network cascade for image super-resolution; vol. 8693 LNCS of *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2014.
22. Wang, Z., Liu, D., Yang, J., Han, W., Huang, T.. Deeply improved sparse coding for image super-resolution. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015:370–378.

23. Dong, C., Loy, C.C., He, K., Tang, X.. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2016;38(2):295–307.
24. Hartley, R., Zisserman, A.. Multiple View Geometry in Computer Vision. Cambridge University Press; 2000.
25. Steger, C.. An unbiased detector of curvilinear structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1998;20(2):113–125.
26. Steger, C.. Unbiased extraction of lines with parabolic and gaussian profiles. *Computer Vision and Image Understanding* 2013;117(2):97–112.
27. Usamentiaga, R., Molleda, J., García, D.F., Pérez, L., Vecino, G.. Real-time line scan extraction from infrared images using the wedge method in industrial environments. *Journal of Electronic Imaging* 2010;19(4).
28. Wedowski, R.D., Atkinson, G.A., Smith, M.L., Smith, L.N.. A system for the dynamic industrial inspection of specular freeform surfaces. *Optics and Lasers in Engineering* 2012;50(5):632–644.
29. Lin, Y., Saripalli, S.. Road detection and tracking from aerial desert imagery. *Journal of Intelligent and Robotic Systems: Theory and Applications* 2012;65(1-4):345–359.
30. Almoussa, N., Dutra, B., Lampe, B., Getreuer, P., Wittman, T., Salafia, C., Vese, L.. Automated vasculature extraction from placenta images. In: *Progress in Biomedical Optics and Imaging - Proceedings of SPIE*; vol. 7962. 2011:.
31. Illingworth, J., Kittler, J.. A survey of the hough transform. *Computer Vision, Graphics and Image Processing* 1988;44(1):87–116.
32. Hough, P.V.S.. Method and means for recognizing complex patterns. US Patent 3069654 1962;.
33. Duda, R.O., Hart, P.E.. Use of the hough transformation to detect lines and curves in pictures. *Communications of the ACM* 1972;15(1):11–15.
34. Ballard, D.H.. Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition* 1981;13(2):111–122.
35. Xu, L., Oja, E., Kultanen, P.. A new curve detection method: randomized Hough transform (rht). *Pattern Recognition Letters* 1990;11(5):331–338.
36. Kälviäinen, H., Hirvonen, P., Xu, L., Oja, E.. Probabilistic and non-probabilistic Hough transforms: overview and comparisons. *Image and Vision Computing* 1995;13(4):239–252.
37. Canny, J.. Computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1986;PAMI-8(6):679–698.
38. Barnard, S.T.. Interpreting perspective images. *Artificial Intelligence* 1983;21(4):435–462.
39. Lutton, E., Maitre, H., Lopez-Krahe, J.. Contribution to the determination of vanishing points using Hough transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1994;16(4):430–438.
40. Shufelt, J.A.. Performance evaluation and analysis of vanishing point detection techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1999;21(3):282–288.
41. Rother, C.. A new approach to vanishing point detection in architectural environments. *Image and Vision Computing* 2002;20(9-10):647–655.
42. Antone, M.E., Teller, S.. Automatic recovery of relative camera rotations for urban scenes. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*; vol. 2. 2000:282–289.
43. Tardif, J.. Non-iterative approach for fast and accurate vanishing point detection. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2009:1250–1257.
44. Pflugfelder, R.. Self-calibrating cameras in video surveillance. Ph.D. thesis; Graz University of Technology, Austria; 2008.
45. Nieto, M., Salgado, L.. Non-linear optimization for robust estimation of vanishing points. In: *Proceedings - International Conference on Image Processing, ICIP*. 2010:1885–1888.
46. Breed, K., Kang, E.Y.. Vanishing point architectural image retrieval. In: *Proceedings of the 2008 International Conference on Image Processing, Computer Vision, and Pattern Recognition, IPCV 2008*. 2008:800–806.
47. Schmitt, F., Priese, L.. Vanishing point detection with an intersection point neighborhood. *Lecture Notes in Computer Science* 2009;5810:132–143.
48. Capel, D., Zisserman, A.. Computer vision applied to super-resolution. In: *IEEE Signal Processing Magazine*; vol. 20. 2003:75–86.
49. Vandewalle, P., Süsstrunk, S., Vetterll, M.. A frequency domain approach to registration of aliased images with application to super-resolution. *Eurasip Journal on Applied Signal Processing* 2006;2006:1–14.
50. Schultz, R.R., Stevenson, R.L.. Extraction of high-resolution frames from video sequences. *IEEE Transactions on Image Processing* 1996;5(6):996–1011.
51. Schultz, R.R., Meng, L., Stevenson, R.L.. Subpixel motion estimation for super-resolution image sequence enhancement. *Journal of Visual Communication and Image Representation* 1998;9(1):38–50.
52. Patti, A.J., Sezan, M.I., Tekalp, A.M.. Superresolution video reconstruction with arbitrary sampling lattices and nonzero aperture time. *IEEE Transactions on Image Processing* 1997;6(8):1064–1076.
53. Otsu, N.. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics* 1979;9(1):62–66.
54. Yuan, X., Wu, L., Peng, Q.. An improved Otsu method using the weighted object variance for defect detection. *Applied Surface Science* 2015;349:472–484.
55. Ng, H.. Automatic thresholding for defect detection. *Pattern Recognition Letters* 2006;27(14):1644–1649.
56. Fan, J., Lei, B.. A modified valley-emphasis method for automatic thresholding. *Pattern Recognition Letters* 2012;33(6):703–708.
57. Seo, K.S., Lee, J.H., Choi, H.M.. An efficient detection of vanishing points using inverted coordinates image space. *Pattern Recognition Letters* 2006;27(2):102–108.
58. Schaffalitzky, F., Zisserman, A.. Planar grouping for automatic detection of vanishing lines and points. *Image and Vision Computing* 2000;18(9):647–658.
59. Mirzaei, F.M., Roumeliotis, S.I.. Optimal estimation of vanishing points in a manhattan world. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2011:2454–2461.