Virtual camera modeling for multi-view simulation of surveillance scenes

Niccoló Bisagno and Nicola Conci
DISI - Department of Information Engineering and Computer Science
University of Trento
Trento, Italy
{niccolo.bisagno, nicola.conci}@unitn.it

***Abstract*—A recent trend in research is to leverage on advanced simulation frameworks for the implementation and validation of video surveillance and ambient intelligence algorithms. However, in order to guarantee a seamless transferability between the virtual and real worlds, the simulator is required to represent the real-world target scenario in the best way possible. This includes on the one hand the appearance of the scene and the motion of objects, and, on the other hand, it should be accurate with respect to the sensing equipment that will be used in the acquisition phase. This paper focuses on the latter problem related to camera modeling and control, discussing how noise and distortions can be handled, and implementing an engine for camera motion control in terms of pan, tilt, and zoom, with particular attention to the video surveillance scenario.**

***Index Terms*—Camera model, PTZ, video surveillance**

I. Introduction

Video surveillance has been a matter of study for the past three decades, and researchers have investigated many different facets of the subject, ranging from simple motion detection and segmentation algorithm [1], [2], to object tracking [3], person re-identification [4], and analysis of complex crowded scenes [5]. Although most video surveillance networks rely on the use of ordinary static cameras, there is an increasing trend in updating the existing infrastructures with smart cooperative networks of cameras. In such a scenario cameras are required to share relevant information over the network, in order to improve the tracking of objects and provide the best possible coverage [6]. To guarantee flexibility and dynamic reconfiguration of the camera network, a viable option is to adopt PTZ (Pan-Tilt and Zoom) cameras, which allow to track and focus on specific objects of interest in the scene thanks to the possibility of dynamically repositioning the sensor [7].

However, researchers in this domain keep facing two common problems: (i) the lack of labeled data, especially for those rare events, i.e., anomalies, that should be detected by the monitoring infrastructure, and (ii) the incapability of reproducing the same type of event when dealing with reconfigurable camera networks. One possible solution to tackle such limitations is to deploy virtual environments and simulation frameworks. Virtualization has been subject of research in the camera networks community [8], [9] and crowd analysis [5].

Most papers in the state of the art adopting simulation, have focused on the deployment and assessment of different network configurations to guarantee a good coverage of the scene thus improving the chances of detecting critical events. However, there is no sufficient literature that demonstrated the transferability of the lessons learned from the simulated environment into the real world. In fact, it is to be noted that when a sequence is recorded using a virtual framework, we must also capture the peculiarities of the specific sensor used for the acquisition. Besides the actual modeling of the simulated environment, also the camera modeling has to be representative of the real equipment, being able to model multiple features, as for example the noise sources and distortions of real cameras and lenses.

In the computer vision field, advanced graphical simulation frameworks are being exploited to perform data augmentation [10]. To our knowledge, the rendering and visual appearance of the scenes has been studied and developed rather thoroughly, while the peculiarities of the virtual recording system have not been deeply investigated. Camera models implemented in modern computer graphics engines aim at producing contents which enhance user quality of experience [11], rather than producing realistic (noisy) images.

To correctly model a camera network in a virtual environment we need to deal with the camera intrinsic and extrinsic parameters. We also need to model different kinds of noise sources and distortions, which are typical of real cameras.

Extending the model to include PTZ cameras, also requires to model the camera motion, which significantly impacts the types of algorithms that can be used for the analysis, in terms of object detection and tracking, since common background subtraction techniques would be impaired by the apparent motion of the background.

In this paper we present a framework for modeling a set parameters and distortions related to lenses and camera sensors. We then focus on the specific case of a PTZ camera, showing how to deal with the challenges posed by the camera motion. Eventually, we present a use case scenario for the developed camera model, deployed in a simulated camera network.

II. Camera model description

As mentioned above, modeling an acquisition system requires taking into account two main components, namely the lens and the image sensor. In the real world, these elements are

ISBN 978-90-827970-1-5 © EURASIP 2018

sources of noise and distortion, making the acquisition process significantly different from the theoretical ideal model. In this section we propose a virtual camera model able to deal with the noise and distortions existing in real devices, and also handling the needs in terms of camera motion and control.

A. Lenses

In optics, a lens is a refractive device, which either focuses or disperses light beams. In order to capture an image, an ideal lens focuses all the captured light on a single point and is characterized by a certain numbers of parameters, as:

- focal length
- field of view (FOV)
- depth of field
- aperture

In our simulation framework we focus on modeling the following distortions:

- chromatic aberration
- radial distortion

1) Focal length and Field of View: The focal length of a lens is a measure of how the incoming light is either diverted or converged. The bigger the focal length, the higher will be the magnification of an object. The relation between the focal length and the magnification of the object is ruled by the *thin lens* equation:

$$\frac{1}{f} = \frac{1}{u} + \frac{1}{v} \tag{1}$$

where f is the focal length, u is the distance between the lens and the object and v is the distance between the focal length and the image plane.

The angular field of view (AFOV) is defined as the maximum angular size of an object of interest that can be captured by the camera. The object under inspection is supposed to be at an infinite distance from the lens:

$$AFOV(°) = 2 * tan^{-1}(\frac{h}{2f}) \tag{2}$$

where h is the horizontal sensor size and f is the focal length in millimeters. Common simulation tools natively offer the FOV as a parameter to be set.

The described model is embedded in our simulation framework to re-project the acquired environment onto the image plane.

2) Aperture: the aperture of a camera regulates the amount of light reaching the image sensor. The aperture size is usually regulated by a device called diaphragm, which increases or decreases the aperture size at a factor of two aperture area per stop. The f-number (N) of a camera lens corresponds to the ratio between the focal length f and the diameter D of the aperture:

$$N = \frac{f}{D} \tag{3}$$

When modeling the aperture in synthetic images, we need to take into account that a smaller value of N causes a wider aperture size (we are allowing more lights to reach the sensor). An higher value of f causes the camera aperture to become narrower, thus allowing less light to reach the sensor.

3) Depth of Field: The depth of field is defined as the distance between the nearest and the further object, located in the zone of acceptable sharpness in a photo. The depth of field is determined by three main factors: focal length, distance of the object from the camera, and aperture.

The f-number controls how wide the depth of field will be around the subject that the camera is capturing. The lower the value, the shallower the total depth of field being captured; the higher the value, the wider the total depth of field.

To model the depth of field, we use the full derivation of formulas presented in [12].

4) Chromatic aberration: Chromatic aberration is the effect caused by the inability of the lens to focus all the different colors in the same point, as shown in Fig. 1.

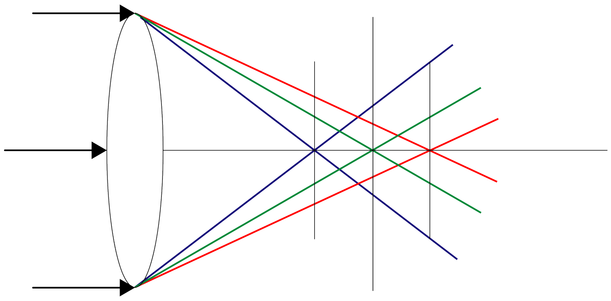


Fig. 1. Chromatic aberration depends on the lens inability to focus the entire color range in the same spot.

In order to simulate the chromatic aberration effect, we then need to slightly separate the color spectrum at the edges and corners of the image.

5) Lens radial distortion: Common lens distortions present or can be approximated as having symmetries along the radial axis. They are usually classified in three different classes: barrel distortion, pincushion distortion, and a combination of the previous two, the so-called mustache distortion.

Barrel distortion, as shown in Fig. 2.a, is characterized by the decrease of the object magnification as the distance from the optical axis increases. This distortion is sometimes intentionally used to obtain the so called fish-eye effect.

Pincushion distortion, as shown in Fig. 2.b, is characterized by the increase of the object magnification as the distance from the optical axis decreases.

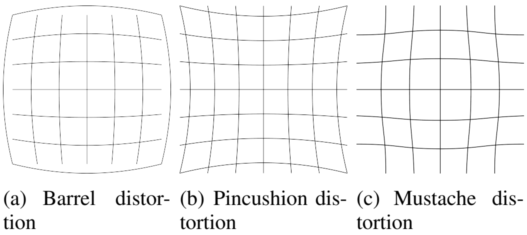
Mustache distortion, as shown in Fig. 2.c, presents a mixture of the two previous distortions. A barrel-like distortion is present toward the center of the image and it becomes a pincushion-like distortion as the distance from the radial axis increases.

The mathematical formulation to correct those distortions is the Brown-Conrady model [13], [14], which has been used in our implementation to test and correct the simulation model we have implemented.

B. Camera sensor

The sensor is the element in a camera, which transforms the incoming light rays into an electrical signal. The signal

ISBN 978-90-827970-1-5 © EURASIP 2018



(a) Barrel distortion (b) Pincushion distortion (c) Mustache distortion

Fig. 2. Common radial distortions patterns in real lenses.

is represented in our case by an image. Since we are trying to reproduce a digital image, we have to understand how to reproduce the characteristics and distortions of a digital sensor. In a camera, the lens and the sensor influence parameters such as the field of view. If the sensor is too small to capture all the light convoyed by the lens, the effective field of view is determined by the sensor. If the captured light does not fill the whole senor area, the effective field of view is determined by the lens. The image sensor format of a digital camera determines the angle of view of a particular lens when used with a particular sensor. The fundamental elements that characterize a sensor are:

- sensor size (format)
- resolution
- dynamic range
- camera sensor noise

1) Sensor size: The sensor size (or sensor format) indicates the shape and the size of the sensor capturing the light. It determines how much light will be used to produce the final image. It determines the final size and format of image that can be captured with a camera device. The size of sensor ultimately determines how much light it uses to create an image. Increasing the sensor size causes the depth of field to decrease, aperture being fixed.

2) Resolution: Resolution of a camera is the ability to distinguish details in the image. It is usually limited by the lens diffraction and by the sensor resolution.

Optical resolution describes the ability of an imaging system to resolve detail in the object that is being imaged. Resolution is usually measured in pixels.

In the simulation framework we are able to manually set the resolution and resize the image as needed.

3) Dynamic Range: The dynamic range is defined as

$$DR = \log \frac{N_{max}}{N_f} \tag{4}$$

where N_{max} represents the maximum signal level that the sensor can output, and N_f represents the noise floor at minimum amplification. The noise floor is calculated as the root mean square of the noise level in a black image. The dynamic range measures the capability of a sensor to capture the brightest and darkest spot in an image and the number of levels in between.

Modeling the dynamic range of a sensor in a simulated environment is achieved starting from the color distribution of an object in a scene and applying some contrast stretch techniques, as also commonly used in photography. Contrast stretching is a technique, which aims at improving/modifying the quality of an image by stretching or compressing the intensity value of the different colors, such that it fits the desired interval of values. In our methodology we are able to set the lower and the upper limit of the stretched histogram, in order to fit the color values of our virtual camera to the one of a real device. Contrast stretch adaptation speed also allows to reproduce common camera effects in videos when there is a sudden change in lighting.

4) Camera sensor noise: Ideally, the camera sensor should produce exactly one electron for each photon striking one of his pixels. In practice, the process, which allows the camera sensor to convert light into a proper image is affected by noise. In captured images, noise can be seen as a granular color variation on surfaces, which look uniform at a distance.

In [15], noise is segmented into spatio-temporal categories to be measured. The final noisy image N_{cap} is defined as

$$\begin{aligned} N_{cap} = (I * PRNU + SN_{ph}(I) + FPN + \\ + SN_{dark} + N_{read}) * N_D * N_{filt} + N_Q \end{aligned} \tag{5}$$

where I is the sensor irradiance, PRNU is the photo response non-uniformity, SN_{ph} is the photon shot noise, FPN is the offset fixed-pattern noise, SN_{dark} is the dark-current shot noise, N_{read} is the readout noise, N_D is the demosaicing noise, N_{filt} is the post image capture effect, and N_Q is the quantization noise.

The distribution of all sources at a given CCD-sampling frequency is measured as an additive Gaussian distribution.

For the simulation, we are interested in reproducing the noise intensity and overall distribution rather then exactly calibrating the model to reproduce a specific camera brand or type. This is achieved by adding a white Gaussian noise, which can be modified in terms of mean value and standard deviation to fit the requirements at hand. It allows the simulation of typical scenarios, such as the noise in low light conditions and bloom borders.

III. Pan-Tilt-Zoom Camera

In visual surveillance, the use of PTZ (Pan-Tilt and Zoom) camera has been thoroughly investigated [8], [9], [1]. PTZ cameras provide an effective way to increase the coverage of a certain area thanks to their ability to move, either by progressively scanning the environment or zooming in to specific locations in presence of events of interest. In a cooperative camera network, PTZ cameras have to be able to dynamically detect and track the objects of interest [1], [7], [2], guaranteeing a smooth handover across cameras.

Therefore, when modeling such cameras, we must deal with the motion parametrization of each camera in the network, thus acting on both the intrinsic and extrinsic camera parameters.

1) PTZ motion model: In this section we describe the model used to replicate the movement of a PTZ camera in a virtual environment. Different types of cameras are available on the market, like mechanical PTZ and virtual PTZ. The term virtual implies that no physical sensor movement occurs; the captured images, instead, are obtained by cropping from the full resolution picture obtained by a high resolution image. Generally, to model the motion of a PTZ camera we need to replicate its three extrinsic parameters (pan, tilt and zoom), assuming that the camera is anchored to a fixed location.

Also pan and tilt are subject to constraints, and the step size for the variation must be defined to control the velocity response of the camera at each time step. Dynamically changing the Field of View of the virtual PTZ allows for zoom modeling, along with providing maximum and minimum range for the FOV to vary.

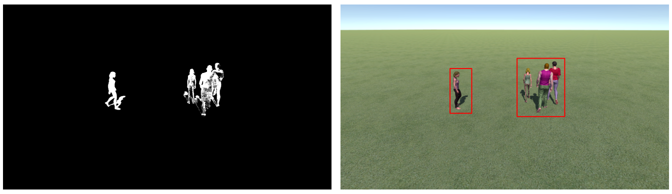
2) PTZ tracking algorithms: The development of a real-time object tracking algorithm to be applied on a PTZ camera, must tackle a variety of problems, such as camera movement, complex object motion, presence of other moving objects in the video scene, and real-time processing requirements.

Algorithms satisfying these constraints can be divided into two main classes: the ones relying on background segmentation [1], [7], and the ones that allow to track only specific objects classes [2].

Algorithms exploiting the background segmentation have to cope with the constant camera movement, requiring re-initialization every time a camera displacement is triggered.

Other algorithms can track even in presence of camera motion, relying on common vision [16], and machine learning [3] tools. However, the main drawback is the difficulty in dealing with the real-time constraints and the related computational costs, which is addressed in literature by either tracking only a category of objects [3] or by developing customized hardware [2].

In our test application we exploit a simple background segmentation application to detect the object, which is then tracked using a state of the art appearance-based tracking, namely the cam-shift algorithm [17], as shown in Fig. 3.


(a)

(b)

Fig. 3. On the original image we perform the background subtraction (a) and blob detection (b). The histogram computed on the blob is then provided as input to the cam-shift algorithm, which is able to track the target even in presence of camera motion.

IV. Results

To validate the simulation of the focal length, we use the Camera Calibration Toolbox [18]. From the conducted experiments we noticed that the simulated focal length differs from the ground truth in millimeters of an average error of 4%.

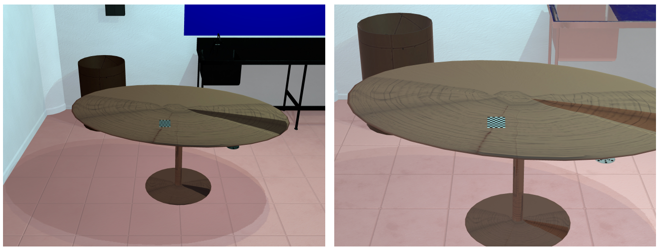
We provide samples of computer generated images which have been recorded using our camera model. In Fig. 4 we show an example of the variation of the focal length on a sample virtual image, without moving the camera. As can be seen, the field of view angle decreases as the focal length increases.

In Fig. 5, we show a noise pattern which increases as a multiplication factor. It is also possible to vary the noise pattern depending on the channel of interest. As can be seen, the noise pattern looks comparable with the noise generated by camera sensors in presence of low illumination, introducing artifacts and color aberrations.

In Fig. 6, we show different examples of distortions. To validate the distortions applied, we applied rectification to restore the undistorted images according to [19].

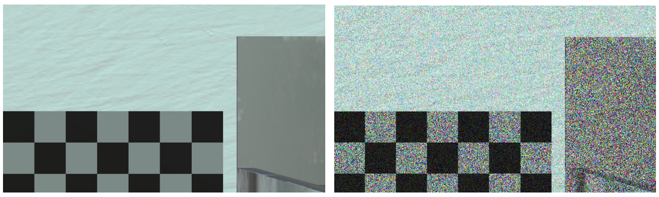
In Fig. 7, we show the effect of different values of contrast stretching in the images.

Similarly to all the other elements that characterize a camera, the developed mode also takes into account the depth of field, which is a crucial parameter that can alter the perception of objects in a visual scene, in terms of sharpness and level of detail. A sample view to show the capabilities of handling the depth of field is shown in Fig. 8.


(a) FL: 25 mm, FOV: 48.1°

(b) FL: 40 mm, FOV: 31.2°

Fig. 4. Example of images taken by a fixed camera (virtual) with an increasing focal length. FL corresponds to the focal length and FOV is the field of view angle.


(a)

(b)

Fig. 5. Detail of a noiseless (a) and noisy (b) image affected by the camera sensor noise.

V. Use case: Camera Network

In camera network research, virtual vision for testing and deploying network has long been a subject of research [8], [9]. Recent improvements in computer graphics have expanded the possibilities of using game engines to tackle computer vision tasks [10]. Deploying a network of cameras in a virtual environment would provide researchers with a valid testbed for validation and benchmarking purposes. Tracking algorithms can

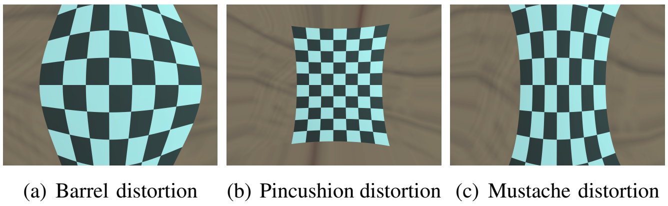


Fig. 6. Example of radial distortions applied in the simulation framework.

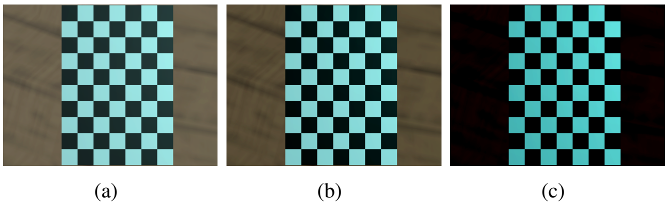


Fig. 7. Modifying the minimum and maximum value of color stretch it is possible to change the appearance of the scene.

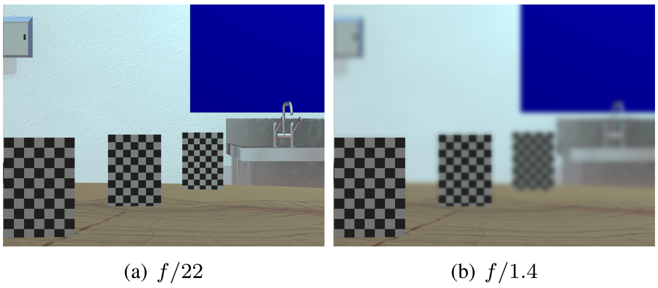


Fig. 8. Examples of synthetic depth of field effect. The images are captured using a fixed camera and varying the aperture. The f-number is reported for each image. The setting consists of a fixed camera with 3 checkerboards at a distance of 1, 1.5, and 2 meters from the camera.

benefit of the ground truth knowledge, which does not need to be manually annotated, and different tracking algorithms can be tested on the very same scene. Also re-identification and tracking of subjects across cameras is another area in which the data recorded in the simulated environment can be effectively exploited. Besides effectively tracking an object, a network should be able to optimize the coverage of the environment. Optimization of the camera deployment is the first step to guarantee the best camera displacement [6]. In case of PTZ cameras, at running time cameras must be able to correctly perform hand-offs. While a camera is focused on tracking a target, the other cameras should be able to reconfigure, so as to guarantee the maximum coverage of the space of interest. This paper represents the starting point for the development of such validation frameworks, tackling the problem of camera modeling in terms of distortions, noise, and PTZ control, through the parametrization of such artifacts within the simulation environment.

VI. Conclusions

In this paper we presented a framework for modeling camera modeling in a simulation framework, highlighting the need of properly handling the issues of noise and distortions. We showed how smart camera networks and PTZ cameras research would benefit from the application of the virtual vision paradigm, relying on sophisticated 3D engines to replicated real challenges in the synthetic domain. Future work will focus on the development of a full framework to allow the testing and evaluation of smart camera networks algorithm, allowing researchers to test their own solutions, for tracking, layout optimization and cooperation among cameras.

References

[1] P. Azzari, L. Di Stefano, and A. Bevilacqua, "An effective real-time mosaicing algorithm apt to detect motion through background subtraction using a ptz camera," *Conference on Advanced Video and Signal Based Surveillance*, pp. 511–516, 2005.

[2] A. Bevilacqua and P. Azzari, "High-quality real time motion detection using ptz cameras," *International Conference on Video and Signal Based Surveillance*, pp. 23–23, 2006.

[3] J. Ahmed, M. Jafri, J. Ahmad, and M. I. Khan, "Design and implementation of a neural network for real-time object tracking," *Proceedings of World Enformatika Conference Machine Vision and Pattern Recognition*, 2005.

[4] S. Messelodi and C. M. Modena, "Boosting fisher vector based scoring functions for person re-identification," *Image and Vision Computing*, vol. 44, pp. 44–58, 2015.

[5] J. C. S. J. Junior, S. R. Musse, and C. R. Jung, "Crowd analysis using computer vision techniques," *Signal Processing Magazine*, vol. 27, no. 5, pp. 66–77, 2010.

[6] K. R. Konda and N. Conci, "Optimal configuration of ptz camera networks based on visual quality assessment and coverage maximization," *International Conference on Distributed Smart Cameras*, pp. 1–8, 2013.

[7] S. Kang, J.-K. Paik, A. Koschan, B. R. Abidi, and M. A. Abidi, "Real-time video tracking using ptz cameras," *International Conference on Quality Control by Artificial Vision*, vol. 5132, pp. 103–112, 2003.

[8] F. Z. Qureshi and D. Terzopoulos, "Surveillance in virtual reality: System design and multi-camera control," *Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2007.

[9] G. R. Taylor, A. J. Chosak, and P. C. Brewer, "Ovvv: Using virtual worlds to design and evaluate surveillance systems," *Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2007.

[10] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," *European Conference on Computer Vision*, pp. 102–118, 2016.

[11] C. C. Bracken and P. Skalski, "Presence and video games: The impact of image quality and skill level," *Annual International Workshop on Presence*, pp. 28–29, 2006.

[12] S. F. Ray, *Applied photographic optics: Lenses and optical systems for photography, film, video, electronic and digital imaging*. Focal Press, 2002.

[13] A. E. Conrady, "Decentred lens-systems," *Monthly notices of the royal astronomical society*, vol. 79, no. 5, pp. 384–390, 1919.

[14] D. C. Brown, "Decentering distortion of lenses," *Photogrammetric Engineering and Remote Sensing*, 1966.

[15] K. Irie, A. E. McKinnon, K. Unsworth, and I. M. Woodhead, "A technique for evaluation of ccd video-camera noise," *Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 2, pp. 280–284, 2008.

[16] F. Porikli, "Achieving real-time object detection and tracking under extreme conditions," *Journal of Real-Time Image Processing*, vol. 1, no. 1, pp. 33–40, 2006.

[17] G. R. Bradski, "Computer vision face tracking for use in a perceptual user interface," 1998.

[18] J.-Y. Bouguet, "Camera calibration tool-box for matlab," *http://www.vision.caltech.edu/bouguetj/calib_doc/*, 2002.

[19] Z. Zhang, "A flexible new technique for camera calibration," *Transactions on pattern analysis and machine intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.