energies

MDPI

Article

2D–3D Spatial Registration for Remote Inspection of Power Substations

Leandro Mattioli *, Alexandre Cardoso and Edgard Lamounier

Faculty of Electrical Engineering, Federal University of Uberlândia, Uberlândia 38408-902, Brazil; alexandre@ufu.br (A.C.); lamounier@ufu.br (E.L.)

* Correspondence: leandro.mattioli@gmail.com or leandro.mattioli@ufu.br

Received: 6 October 2020; Accepted: 18 November 2020; Published: 25 November 2020

Abstract: Remote inspection is critical for smart factories, power systems and undersea and space exploration, among other domains. However, these applications have conflicting requirements: operators should experience high situation-awareness, implying a considerable amount of data to be presented, while having a minimal sensory load, not to compromise the time to make decisions. Recent research suggests computer vision inspection and the adoption of virtual reality (VR) as an alternative to traditional SCADA interfaces. Nevertheless, although VR may provide a good representation of a substation's state, it lacks some real-time information, available from online field cameras and microphones. This work discusses a method to augment virtual environments of power substations with field images, enabling operators to promptly see a virtual representation of the inspected area's surroundings. In addition, the system interacts with a SCADA database, continuously comparing the equipment states against the ones inferred by processing the field images. Whenever a discrepancy is found, a virtual camera can be teleported to the affected region, speeding up system reestablishment. Our results concern the registration accuracy and performance impact for a simple scenario. The collected metrics suggest good registration levels and low impact on real-time rendering performance.

Keywords: substation automation; SCADA; remote monitoring; registration error; augmented virtuality

1. Introduction

In an industrial context, control room panels have evolved from LEDs and gauges to computer screens with windows, associated with custom layouts and computer graphics animations. More recently, electrical power substations have benefited from the virtual reality (VR) technology by exploring the potential of this advanced user interface to complement the usual single-line diagrams [1]. Since power systems are critical, one may not always rely solely on the state reported by the supervisory control and data acquisition (SCADA) sensors.

Some failures need a quick visual inspection for better diagnostics. To safely allow the site to be unmanned, remote visual inspection, called Remote Inspection (RI) henceforth, becomes a valuable technique. Referring to the SCADA integration, it is possible to check the network's reported state for a disconnect switch against the one inferred from the last image [2,3], acquired by an RI system.

However, traditionally, such inspection systems demand a high level of diffuse attention from the user, who needs to visualize and analyze images in multiple screens or windows. The operator can be "easily overwhelmed with the task of integrating these varied forms of data into a complete global view and understanding of a scene" [4]. In this manner, one reasonable option to be considered is the integration of RI data with the SCADA user interface. To be aligned with the concept of cyber-physical systems

(CPS), one of the main features of Industry 4.0 [5], this integration can be done to the virtual environment associated with the factory [6,7]. Considering the case of power substations, a field image, as captured by a RI system, can be surrounded by a tridimensional model of the nearby "as-built" structure, providing more contextual information and extending its scope. The problem of inserting an image so that their objects match their counterparts in the virtual environment is known as 2D–3D spatial image registration [8], and it is one of the key techniques for augmented virtuality (AV) systems [9]. Recent research work has already suggested that this technique significantly improves operator situation-awareness, especially when the camera views are unintuitive or limited [10]. Telemetry information can also be displayed along with the virtual environment.

While many studies on AV have been published, with fields ranging from surgery [11,12] to gaming [13] and teleoperation [10], so far, none of such AV systems is aware of SCADA states. Although these works mainly concern 2D–3D registration, the captured images could be processed to infer important data. In contrast, substation inspection systems, based on image processing techniques [14], are still not integrated with their digital twins' counterparts (virtual power substation virtual environments).

This work proposes and evaluates a novel way of integrating these technologies, combining VR with on-line images and SCADA data in a single solution. It allows VR applications to query the last known color or thermal images for a given set of regions of interest, providing 2D–3D registration for these images. We evaluate the particular case of disconnector switches' images, which are processed by an existing state inference machine based on computer vision techniques [15]. The inferred states are then compared against the ones reported by SCADA. Whenever there is a discrepancy between these states, the system teleports the VR camera and triggers an alarm. This allows for a quicker system reestablishment routines and failure diagnostics. Viewing the field state from inside the operations center improves safety and reduces costs: a local operator is no longer needed just to confirm whether the power disconnector has opened or closed after a teleoperation command. Effectively, for tele-assisted installations, unnecessary travel is reduced, implying quicker and cheaper reestablishment.

The main objective of this research is to improve power substations with RI by applying augmented virtuality techniques and to demonstrate that this approach is not only feasible but also viable. In this sense, the following specific objectives are enumerated:

1. to identify the barriers related to real-time 2D–3D registration for RI uses, considering a scenario with multiple image sources and multiple virtual environments to be augmented;
2. to describe the registration process in terms of mathematical manipulations and pose estimation algorithms;
3. to assess the quality of the registration according to some quantitative metrics, considering not only if the virtual camera's pose matches exactly the image capture conditions but also some poses with small variations (different points of view); and
4. to provide an architecture capable of integrating field images with SCADA, so that states from these two sources can be checked continuously.

The remaining of this text is structured as follows. Section 2 presents some recent work concerning RI for power substations and some other environments. Some systems featuring 2D–3D registration are presented as well. Section 3 gives some background on the mathematical operations for camera pose estimation, along with the method used for the spatial registration. A method for defining the focal length scale factor is presented in Section 4. The system architecture and its implementation's usage are presented in Section 5. The results are explained in Section 6 and discussed in Section 7, along with some conclusions.

2. Related Work

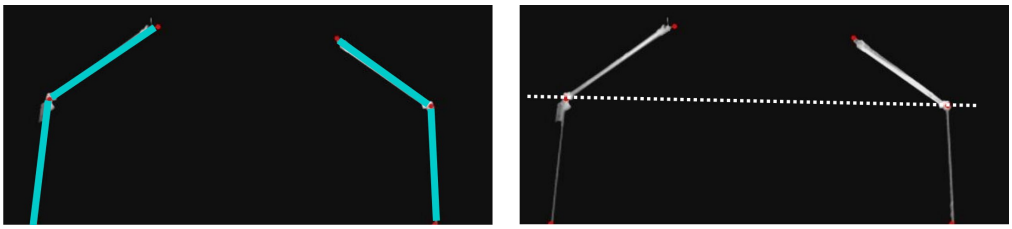
Augmented virtual environments have already been used in urban video monitoring applications [4]. Hu et al. [16] proposed the control of virtual humanoid models' positions, according to real humans' positions captured by video cameras, for outdoor environments. For representative purposes, sending only positions and orientations instead of full images is an interesting strategy, since it demands much fewer network resources, comparing to real-time video streaming. Nonetheless, visual inspection is often needed for better comprehension of the problems in the inspected area.

The 2D–3D registration techniques may also be applied in systems with multiple cameras. Wu et al. [17] proposed a framework for the fusion of large-scale surveillance images with an associated virtual environment. The system combines, in the same view: (i) a mosaic with images captured by the surveillance cameras; (ii) an overall image sent by satellite equipment; and (iii) the corresponding tridimensional model.

When multiple cameras are installed in far remote locations, a guided tour in the monitored environment is particularly interesting to aid operation. Scene-graphs might be put in place for this situation [18]. However, some faults need immediate attention, so waiting for the tour to complete a loop is not an option.

Another application for inserting physical reality information into virtual environments is proposed for the supervision of marine systems. The system deals with the problem of developing a SCADA VR-based interface that reduces sensory overload and "provides situation awareness while maintaining operator capabilities" [19]. However, only a single remote environment is monitored and field images are not available for further visual inspection.

Concerning power tele-assisted substations, video monitoring systems with automatic image analysis are important inspection tools. Color images can be submitted to algorithms capable of detecting people [20], fire [21,22], people climbing ladders in forbidden areas [23], oil leakage in power transformers and unwanted objects left in their nearby [24]. Pereira et al. [15] proposed a way of inferring disconnector switches states by processing their images. The method consists of: (i) extracting a region of interest, comprising the mobile parts of the device and the axes supporting them; (ii) applying a threshold, so that the background is removed; (iii) establishing line equations through linear regression; and (iv) checking the deflection angles to judge as either opened or closed (Figure 1). The solution proposed in our work uses the images and the inferred states from their system.



(**a**) (**b**)

Figure 1. Checking deflection angle (adapted from [15]): (**a**) lines detection; (**b**) establishing base plane and computing deflection angle.

Nevertheless, these systems lack 2D–3D registration for context-aware interpretation, as well as SCADA integration for the detection of telemetry errors.

One common strategy for the detection of irregularities is the foreground–background segmentation. Image segmentation is the computer vision process in which an image is "broken into some non-overlapped

meaningful regions" [25]. In particular, the foreground–background segmentation is, for videos, the separation of what is moving from what is static [26], whereas, for images, the system queries a historical image database to define what is considered to be the image background [24]. This technique can be used to detect motion near a device or to detect objects that, according to historical data, should not be there. Some researchers even suggest the adoption of actions such as deactivating remote control in an area whenever an object motion is detected there [27] (p. 64).

Thermal images are equally important, since "the thermal effect of power devices is one of the major reasons leading to faults" [28]. Drones with thermal cameras have already been deployed to scan faults in substations, storing pictures of insulators, which are later processed for failure diagnostics [14]. In addition to all security issues related to drones in substations, this approach has the same limitations considering SCADA integration and situation awareness.

Alternatively, field images can be augmented with thermal sensor data [29]. However, since telemetry data are error-prone, in this case, the inspection will be restricted to failures visible in the captured color images. It should be noted that, even with this limitation, the proposed system is integrated to SCADA data with video monitoring. Augmented reality can also be used by field operators to better visualize contextual SCADA data [30], although the scope of this work is the opposite integration: having more field information in the operations centers.

Equipment inspection can also be realized by inspector robots [31]. Considerably more complex, these systems combine: (i) the needed advanced techniques for the design of autonomous robots, such as route planning, collision detection, battery management, environment mapping and information fusion; (ii) machine learning; and (iii) failure detection employing computer vision routines.

Finally, when dealing with cameras equipped with pan–tilt–zoom (PTZ) control, a common scenario is to capture multiple views, multiplexed in time, so that more than one asset can be surveilled. This approach suffers from a limitation: servomotors' motion, periodically changing their setpoints to allow different poses, generate cumulative errors that must be constantly compensated. Online camera calibration has been already evaluated for substation video monitoring [32].

Nevertheless, none of these works provides, at the same time: (i) support for multiple installations and cameras, including both thermal and color images; (ii) integration with SCADA; and (iii) optimal spatial registration without the need of in loco camera calibration. Calibrating many remote cameras by taking pictures of the checkerboard pattern from many angles [33] would be impractical to the operations center. Although requiring all sites to use the same camera device (or just a few models) could be an alternative, this would make the system too restrictive in terms of compatibility. Therefore, optimizing and inferring intrinsic parameters solely from the inspection images is an important feature for this kind of application. Besides, a SCADA integration, combined with the spatial registration, can significantly improve the user interface for RI systems, speeding up the visualization and contextualization of the faults inferred from the image processing routines cited above.

3. 2D–3D Spatial Registration Formulation

This section discusses the problem of inserting images acquired from field cameras into the associated virtual environments. An important requirement is to match some objects or points in the image with their correspondences in the tridimensional model. If neither the real nor the virtual camera has significant distortion and skew, one possible method for this kind of spatial registration is to estimate the camera pose, considering the field camera's image and the intrinsic parameters of the virtual camera. This can be done by iterative and analytical methods. The overall registration quality directly affects the operator's interpretation speed and is tightly related to the accuracy of the estimated pose, thus making this estimation

crucial for RI. In addition, the large-scale scenario with multiple remote environments and cameras imposes additional constraints, needing fast spatial registration, not to compromise real-time rendering.

In this section, all vectors are column vectors and thus are transposed when displayed inline in the text.

3.1. Coordinate Systems

The perspective transformation and the spatial registration provide mappings between the image space and world space. The former relates to pixels coordinates, whereas the latter is defined based on a global Cartesian coordinate system [34] (p. 77).

The following coordinate systems are used in this text.

Image homogeneous coordinate system (ICS):

In this coordinate system, a point $Q = (q_x, q_y, \lambda)$, with $\lambda \neq 0$, refers to a pixel in the image, located at $(q_x/\lambda, q_y/\lambda)$. The vectorial function $\eta(\boldsymbol{g})$ is used in this text to denote homogeneous coordinates normalization (1):

$$\eta(\boldsymbol{g}) = \frac{1}{\lambda} \cdot \boldsymbol{g} = \begin{bmatrix} g_x & g_y & 1 \end{bmatrix}^T. \tag{1}$$

World coordinate system (WCS):

The world positions are described with a standard right-handed (counter-clockwise) coordinate system, with the z-axis in the vertical direction.

Game engine's coordinate system (GCS):

The software package used for composing the substation scene, namely Unity 3D [35], has a left-handed (clockwise) coordinate system with the y-axis in the vertical direction. Therefore, all results expressed in WCS must still be transformed to the engine's coordinate system.

3.2. Perspective Projection Transformation

The perspective projection is a particular kind of linear transformation, capable of mapping points from world space to their correspondents in image space. Let $P = (p_x, p_y, p_z)$ be a point defined in world space and $Q = (q_x, q_y, \lambda)$ be the homogeneous coordinates of the pixel that is the result of the perspective projection of P in the image plane.

Considering the finite projective camera model [36] (pp. 154–157), this transformation can be stated as:

$$\begin{bmatrix} q_x \\ q_y \\ \lambda \end{bmatrix} = \boldsymbol{C} \left[\boldsymbol{R} \middle| \boldsymbol{t} \right] \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix}, \tag{2}$$

where $\boldsymbol{C}$ is the 3×3 matrix of the camera intrinsic parameters, explained below and $[\boldsymbol{R}|\boldsymbol{t}]$ is the 3×4 joint rotation–translation matrix divided up into the 3×3 rotation matrix $\boldsymbol{R}$ plus the translation vector $\boldsymbol{t}$.

The camera matrix $\boldsymbol{C}$ is given by:

$$\boldsymbol{C} = \begin{bmatrix} f_x & \tau & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

where (f_x, f_y) are the focal lengths, τ is the skew coefficient between the x and the y axis and (c_x, c_y) is the optical center (principal point).

Some simplifications might be applied to special cases [36] (pp. 154–157). If pixels are squares, we can consider $f_x = f_y$. If there is no skew effect, then $\tau = 0$. Besides, if the origin of the image coordinate system is located precisely at the image center, then $c_x = c_y = 0$.

3.3. Camera Pose Estimation

Let $\mathbb{I}$ be the set of image pixels and $\mathbb{W}$ be the set of world space points. We are interested in a set of n points in the image, $\{Q_i \in \mathbb{I} | 1 \leq i \leq n\}$, and n points in world space, $\{P_i \in \mathbb{W} | 1 \leq i \leq n\}$, such that for each i there is a unique correspondence $(P_i \mapsto Q_i)$. Stated another way, suppose we have both image homogeneous coordinates (ICS) and their related world space coordinates (WCS) for some keypoints.

The problem of estimating the joint rotation–translation matrix, $[\boldsymbol{R}|\boldsymbol{t}]$, from the keypoints and the camera's intrinsic parameters $\boldsymbol{C}$ is called Perspective-n-Point [37]. This is particularly useful for establishing mappings $(P_i \mapsto Q_i), \forall P_i \in \mathbb{W}$ and $\forall Q_i \in \mathbb{I}$, that is, not only for the keypoints, but also for all other resulting image pixels.

Equation (2) can be reorganized splitting the joint rotation-translation matrix, $[\boldsymbol{R}|\boldsymbol{t}]$, and adjusting the matrices dimensions:

$$\begin{bmatrix} q_x \\ q_y \\ \lambda \end{bmatrix} = \boldsymbol{C} \cdot \boldsymbol{R} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} + \boldsymbol{t}. \tag{4}$$

Once the pose is estimated, Equation (4) can be used to evaluate the homogeneous coordinates of the image pixel, given the coordinates of the point in world space.

The pose estimation is especially interesting for augmented and mixed reality applications since it allows the computation of a virtual object's pose in the image coordinate system.

However, even if the intrinsic parameters are unknown, they can still assume values, due to some further simplifications, as explained in Section 3.2, treating the camera as if it were almost ideal. In such a scenario, distortions are ignored. The principal point is defined in the image center and the focal length elements on the camera matrix assume the same value, proportional to one of the image dimensions.

César et al. [38] compared several Perspective-n-Point (PNP) algorithms, revealing the techniques known as Efficient Perspective-n-Point Camera Pose Estimation (EPnP) [39] and Pose from Orthography and Scaling with Iterations (POSIT) [40] as the most robust. Both methods require the coordinates of four or more non-coplanar keypoints in the virtual world space and their corresponding coordinates in the image. The EPnP brings a non-iterative solution, of complexity O(n), from the evaluation of a weighted sum of eigenvectors of a 12×12 matrix and the solution of a constant number of quadratic equations to adjust the weights. In contrast, the POSIT technique first estimates the object's pose by solving a linear system. After this first estimation, the algorithm enters a loop where the parameters from a previous iteration are used to re-calculate the keypoints projections, which will be used instead of the original ones to repeat the pose estimation, resulting in a, presumably, more accurate result. Recently, a new method for obtaining these camera parameters without having points' correspondences has been proposed [41]. The system described in this paper uses the iterative PNP solver offered by the OpenCV library [42].

3.4. Rectangular Region and Virtual Camera Parameters

Section 3.3 described the problem of estimating the camera pose, allowing the retrieval of the complete transformation from world space to image space. Stated another way, the method enables the mapping $(P_i \mapsto Q_i)$. The inverse problem, i.e., going from the image space to the world space, is used for augmented virtuality systems and can be done analytically, once the camera pose is estimated, as shown in this section. It should be noted that more than one camera model can fulfill this mapping by different poses, depending on the focal length elements in (3). It is desired, however, to set the image as a texture for a rectangular region

overlaid in the virtual environment and to teleport the virtual camera to a pose that is compatible with the VR camera intrinsic parameters, so that the photo and its surrounding virtual environment match optimally.

Let $\boldsymbol{p} = \begin{bmatrix} p_x & p_y & p_z \end{bmatrix}^T$ represent a point P_i in WCS and $\boldsymbol{\mathcal{Q}} = \begin{bmatrix} q_x & q_y & \lambda \end{bmatrix}^T$ represent a point Q_i in ICS. It is straightforward manipulate (4) to isolate the $\boldsymbol{p}$:

$$\begin{aligned} & \boldsymbol{C}^{-1} \cdot \boldsymbol{\mathcal{Q}} = \boldsymbol{C}^{-1}\boldsymbol{C} \cdot \boldsymbol{R} \cdot \boldsymbol{p} + \boldsymbol{t} \\ \Rightarrow \quad & \boldsymbol{C}^{-1} \cdot \boldsymbol{\mathcal{Q}} - \boldsymbol{t} = \boldsymbol{R} \cdot \boldsymbol{p} \\ \Rightarrow \quad & \boldsymbol{R}^{-1} \cdot \left(\boldsymbol{C}^{-1} \cdot \boldsymbol{\mathcal{Q}} - \boldsymbol{t} \right) = \boldsymbol{R}^{-1} \cdot \boldsymbol{R} \cdot \boldsymbol{p} \\ \Rightarrow \quad & \boldsymbol{p} = \boldsymbol{R}^{-1} \cdot \left(\boldsymbol{C}^{-1} \cdot \boldsymbol{\mathcal{Q}} - \boldsymbol{t} \right) \end{aligned} \tag{5}$$

Finally, let $\boldsymbol{q}$ be defined in ICS to represent the same pixel as $\boldsymbol{\mathcal{Q}}$, such that:

$$\boldsymbol{q} = \eta \left(\boldsymbol{\mathcal{Q}} \right) = \frac{1}{\lambda} \boldsymbol{\mathcal{Q}}. \tag{6}$$

Then, the right-hand side can be reorganized in the following two terms:

$$\boldsymbol{p} = \underbrace{\boldsymbol{R}^{-1} \cdot \boldsymbol{C}^{-1} \cdot \boldsymbol{q}}_{\boldsymbol{a}} \cdot \lambda - \underbrace{\boldsymbol{R}^{-1} \cdot \boldsymbol{t}}_{\boldsymbol{b}} \tag{7}$$

Since the scalar λ is not bound to any specific value, the solution set corresponds to a line ℓ, in its parametric form:

$$\boldsymbol{\ell} = \begin{bmatrix} a_x \cdot \lambda - b_x \\ a_y \cdot \lambda - b_y \\ a_z \cdot \lambda - b_z \end{bmatrix}. \tag{8}$$

Indeed, many points $\{P_i\}$ in world space may result in the same projection Q_i.

Image Overlay

The action of overlaying a photo into a virtual environment, for Augmented Virtuality applications, requires determining the pose (position and orientation) of the rectangular region that will display the image in the virtual environment.

Let v_1 be an image of size $w_1 \times h_1$ dots and v_2 be the rectangular region of size $w_2 \times h_2$ in world units. Considering that the virtual environment has high geometrical fidelity, we must first assert that the target rectangular region dimensions match the image storage aspect ratio, that is, $w_1/h_1 = w_2/h_2$.

Now, from the set of all possible lines extracted from (8), by setting values for $\boldsymbol{q}$ in (7), let us consider ℓ_{00}, ℓ_{w0}, ℓ_{0h}, ℓ_{wh}, obtained by using the image vertices. Finally, let ℓ_{cc} be the line obtained by using the image center point. All these lines, as well as all other lines obtained by (5) intercept at $-\boldsymbol{b}$, which corresponds to the estimated position of the camera (Figure 2). Indeed, a quick inspection on (7) reveals that the term $-\boldsymbol{b}$ is equal to $-\boldsymbol{R}^{-1} \cdot \boldsymbol{t}$, thus it does not depend on the image point Q.

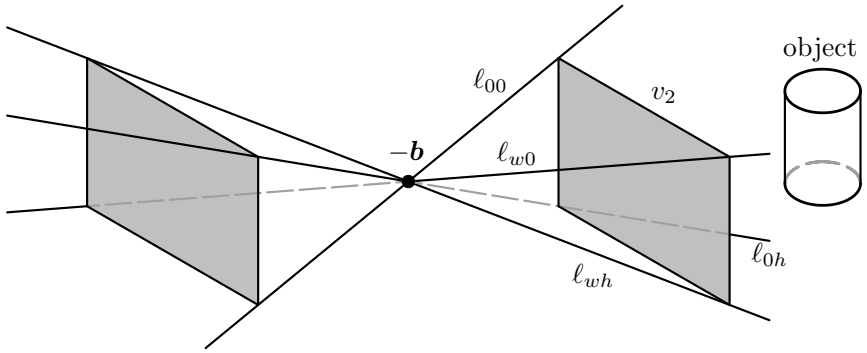


Figure 2. Converging lines and rectangular area v_2.

The result is a right bipyramid with a rectangular base and the apex vertex located at $-\boldsymbol{b}$. The relevant pyramid, associated with the camera's viewing frustum, is the one that contains v_2. Note that this pyramid's base should have the smallest distance to the virtual object. The other pyramid's direction is opposite to the camera and thus is ignored by the solution.

From the known variables and expressions, it is possible to determine the pyramid's height h_3, as shown in Figure 3.

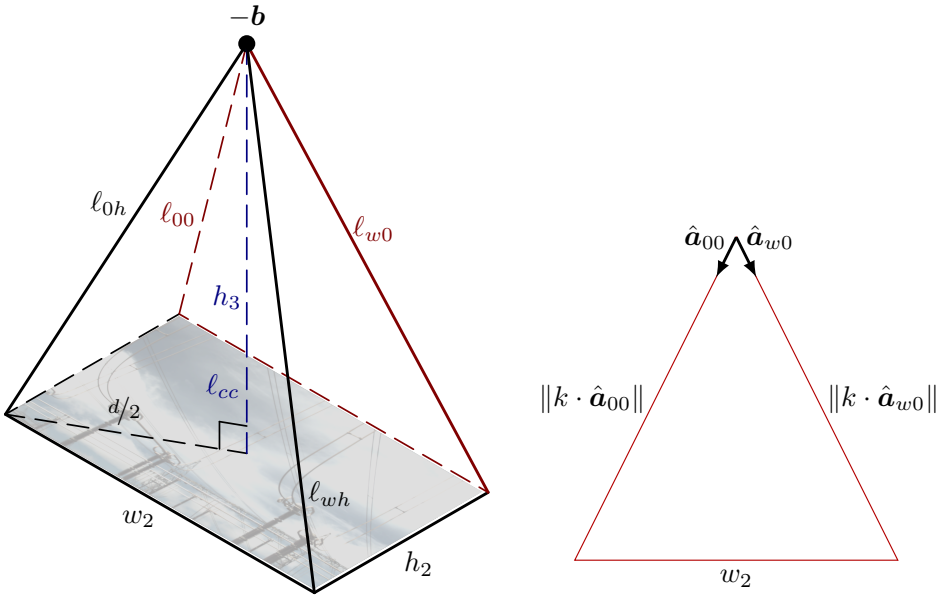


Figure 3. Elements for calculating h_3.

Let ℓ_{ij} denote a line extracted from (8), by using some image point Q specified as the position vector $\boldsymbol{q}$ in (7). In addition, let $\boldsymbol{a}_{ij}$ be the direction vector for line ℓ_{ij} and $\hat{\boldsymbol{a}_{ij}}$ be the unit vector obtained from $\boldsymbol{a}_{ij}$. The coordinates of $-\boldsymbol{b}$ are also known, from any line ℓ_{ij}. The target rectangular region dimensions, w_2 and h_2, are also specified. Finally, the pyramid lateral faces are isosceles triangles. It is easy to determine the value of the scalar k (9).

$$\begin{aligned} \|k \cdot \hat{\boldsymbol{a}}_{00} - k \cdot \hat{\boldsymbol{a}}_{w0}\| &= w_2 \\ \therefore k &= \frac{w_2}{\|\hat{\boldsymbol{a}}_{00} - \hat{\boldsymbol{a}}_{w0}\|} \end{aligned} \tag{9}$$

Since the diagonal d of the rectangular base is given by $d = \sqrt{w_2^2 + h_2^2}$, the pyramid's height h_3 can be equally obtained with the Pythagorean theorem:

$$\|k \cdot \hat{\boldsymbol{a}}_{00}\|^2 = h_3^2 + \left(\frac{d}{2}\right)^2, \qquad h_3 > 0$$
$$\therefore h_3 = \sqrt{\|k \cdot \hat{\boldsymbol{a}}_{00}\|^2 - \left(\frac{d}{2}\right)^2} \tag{10}$$

Then, it is possible to determine the v_2 parameters needed to put it exactly in the pose that the photo was captured, namely the position vector $\boldsymbol{u}$ (referring to the point U) and the coordinate axes $\boldsymbol{\theta}_x$, $\boldsymbol{\theta}_y$ and $\boldsymbol{\theta}_z$ (Figure 4).

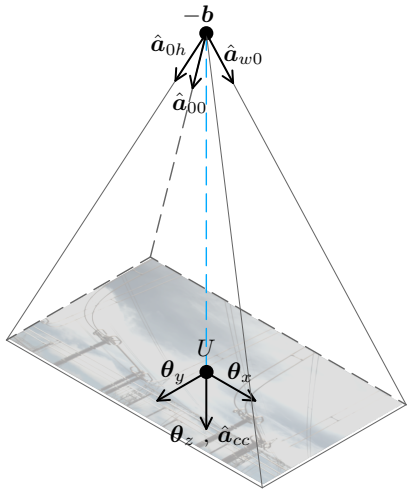


Figure 4. Determining the pose of the rectangular region v_2.

The direction vector of ℓ_{CC} is normal to the plane that contains v_2 and parallel to the pyramid axis, defining the direction $\boldsymbol{\theta}_z$. The other directions can be easily calculated using the unit vectors from the edges. The following equations show all four parameters:

$$\boldsymbol{u} = -\boldsymbol{b} + h_3 \cdot \hat{\boldsymbol{a}}_{cc} \tag{11}$$
$$\boldsymbol{\theta}_x = \frac{\hat{\boldsymbol{a}}_{w0} - \hat{\boldsymbol{a}}_{00}}{\|\hat{\boldsymbol{a}}_{w0} - \hat{\boldsymbol{a}}_{00}\|}, \tag{12}$$
$$\boldsymbol{\theta}_y = \frac{\hat{\boldsymbol{a}}_{0h} - \hat{\boldsymbol{a}}_{00}}{\|\hat{\boldsymbol{a}}_{0h} - \hat{\boldsymbol{a}}_{00}\|} \tag{13}$$
$$\boldsymbol{\theta}_z = \hat{\boldsymbol{a}}_{cc} \tag{14}$$

It should be noted that the unit vectors $\hat{\boldsymbol{\theta}}_x$, $\hat{\boldsymbol{\theta}}_y$ and $\hat{\boldsymbol{\theta}}_z$ are, in fact, the rows from the estimated $\boldsymbol{R}$ matrix. Indeed, the plane containing the rectangular region and the estimated camera have the same orientation.

That way, the pose of v_2 is determined so that the virtual environment can represent the conditions in which the image was taken. The virtual environment camera is positioned at $-\boldsymbol{b}$ and oriented according to the $\boldsymbol{R}$ matrix. Besides eventual distortions and other non-ideal parameters, the real camera is likely to have a different field-of-view (FoV) from the virtual camera. Hence, the camera matrix is an important source of registration errors. Combined with the Perspective-n-Point solver errors, this can lead to bad

quality results. These errors can be reduced by either calibrating the camera or applying some optimization algorithms. The latter approach is discussed in the next section.

4. Focal Length Scale Autoset Method

The RI cameras used in this work have either no mobility or fixed position presets, which is why the keypoints' coordinates are static.

A good consequence of this constraint is that there is no need for running the Perspective-n-Point algorithm for every new image acquired. Once the VR environment knows the parameters for well positioning the overlay image and the virtual camera, the real-time operation consists only of fetching the new images and updating the overlay (a texture).

Taking into account the difficulties of having and maintaining intrinsic parameters calibration for each remote camera of the system, it is reasonable to consider methods that do not require such procedures.

Thus, an iterative method for discovering the optimal focal length $f_x = f_y$ was applied. Once good results are achieved, the parameters' values are stored in a database.

The values for f_x and f_y are obtained by multiplying the image height, h_1, by some scale factor $f_{x,y}$, mapping from pixels to meters. The algorithm's goal is to find the optimal value for $f_{x,y}$ inside a numeric range.

Our method uses a mean distance metric for the objective function (to be minimized) and a ternary-search variant.

Let us recall the symbols used in Sections 3.2 and 3.3, considering a set of keypoints coordinates defined in the image, $\{\boldsymbol{q}_i \in \mathbb{P} \,|1 \leq i \leq n\}$ and in the world $\{\boldsymbol{p}_i \in \mathbb{W} \,|1 \leq i \leq n\}$, such that $(\boldsymbol{p}_i \mapsto \boldsymbol{q}_i)$. Using the homogeneous coordinates normalization function (1), the mean Euclidean distance can be formally stated as:

$$\overline{d_{PNP}} = \frac{1}{n} \sum_{i=1}^{n} \left\| \eta\left(\boldsymbol{q}_i\right) - \eta\left(\rho_f\left(\boldsymbol{p}_i\right)\right) \right\|, \tag{15}$$

where n is the number of keypoints, $\boldsymbol{q}$ is a vector with key point coordinates in the image coordinate system and $\rho_f\left(\boldsymbol{p}_i\right)$ is the result of the perspective projection of the point located at $\boldsymbol{p}_i$ in the overlay rectangular region, considering the camera pose obtained with $f_x = f_y = f_{x,y} \cdot h_1$.

The goal of Algorithm 1 is to find the optimal $f_{x,y} \in [f_{min}, f_{max}]$, such that $\overline{d_{PNP}}$ is minimum:

$$f_{x,y} = \arg\min \frac{1}{n} \sum_{i=1}^{n} \left\| \eta\left(\boldsymbol{q}_i\right) - \eta\left(\rho_f\left(\boldsymbol{p}_i\right)\right) \right\| \tag{16}$$

Algorithm 1 Ternary-search focal length scale factor optimization

procedure FINDOPTIMALF(f_{min}, f_{max})
  $f_1 \leftarrow f_{min}$, $f_4 \leftarrow f_{max}$
  while $(f_4 - f_1 > \epsilon)$ or iterations limit reached **do**
    $f_2 \leftarrow f_1 + (f_4 - f_1)/3$
    $f_3 \leftarrow f_1 + 2 * (f_4 - f_1)/3$
    $error_2 \leftarrow \overline{d_{PNP}}$ metric using values R_2 and T_2
    $error_3 \leftarrow \overline{d_{PNP}}$ metric using values R_3 and T_3
    if $error_2 < error_3$ **then**
      $f_4 \leftarrow f_3$
    else
      $f_1 \leftarrow f_2$
    end if
  end while
  if $error_2 < error_3$ **then**
    return f_2
  else
    return f_3
  end if
end procedure

In each iteration, the search range is subdivided into four uniformly spaced values $f_1, \ldots, f_4$, and then the metric is evaluated in the intermediate points f_2 and f_3, giving conditions to narrow the search range to either $[f_1, f_3]$ or $[f_2, f_4]$. The algorithm runs with a fixed number of iterations, as shown above, or until a considerably small error is found.

5. System Architecture and User Interface

This section describes the system overall architecture, the database model for the field images, and the user interface elements.

The solution deals with one or more substations with one or more cameras. Each camera is related to a single asset of interest (circuit breaker, power switch or transformer). The relation is one-to-many: although many cameras can be associated with the same asset, one single camera does not observe more than one asset. Each substation collects images and other sensory data, sending the former to an image database and the latter to the SCADA system. Both targets are located in a remote operations center. The remaining nodes of the system architecture are the registration server and clients. The registration server is used to handle 2D–3D registration requests, caching results to improve performance. The architecture is depicted in Figure 5.

The registration web server was developed in Python 3, using the Flask micro-framework and the SQLAlchemy object-relational mapper (ORM) [43]. The VR clients were made with the Unity 3D game engine [35]. Finally, the image database was emulated with an SQLite [44] file associated with a dataset of field photos.

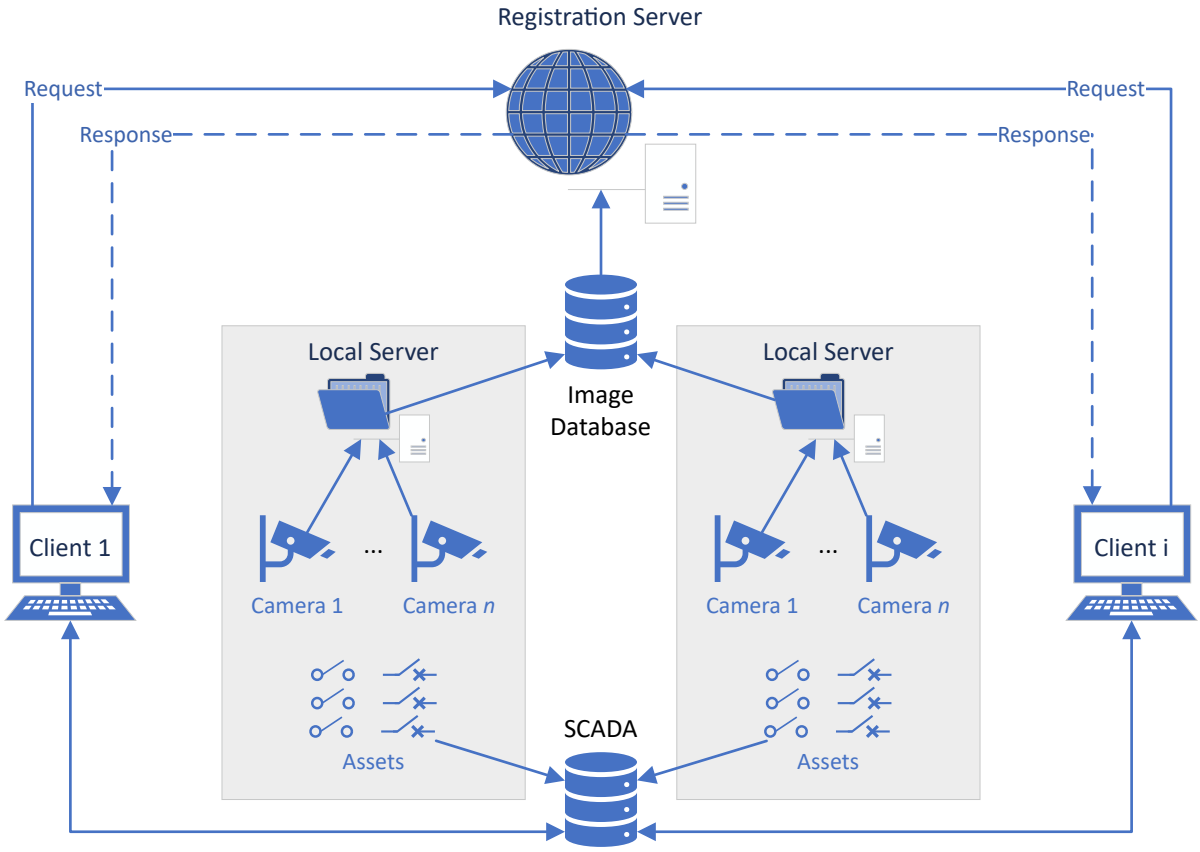


Figure 5. Augmented virtuality system architecture.

5.1. Database Model

The augmented virtuality server stores data referring to the photos fetched from the image database and the 2D–3D registration metadata. A simplified data model is presented in Figure 6.

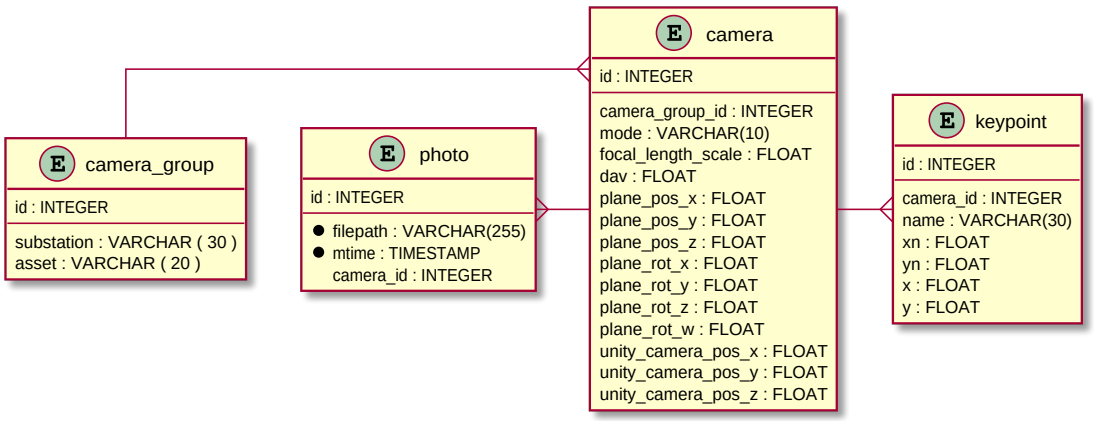


Figure 6. Backend simplified database model.

Each substation has zero or more camera groups, which are sets of fixed-positioned cameras aimed at the monitoring of one asset. A single camera with different position presets, multiplexed in time, is treated as if it were multiple fixed cameras.

For each camera, a set of keypoints is defined. These are pixel coordinates (x, y) and normalized coordinates (x_n, y_n) of the points Q_i from the mappings $(P_i \mapsto Q_i)$ explained in Section 3.3.

Finally, the optimal intrinsic parameters and the results of the 2D–3D registration (pose and quality metric) are stored within the camera records. The plane rotation is stored as a quaternion $(\omega, \begin{bmatrix} x & y & z \end{bmatrix})$, thus using four floating-point fields in the database. Considering fixed cameras, this means that the Perspective-n-Point problem does not need to be executed on each request, but only during the calibration process.

5.2. SCADA Integration

The SCADA database stores both analog values, such as the voltage in a transformer, and digital (i.e., "open" or "closed") device states. The latter are used in our solution, specifically for the case of disconnector switches.

In this sense, the VR environment application makes HTTP requests to a middleware, periodically fetching a report with the digital states of all disconnector switches. For our RI goals, this telemetry data is combined with the on-line images, providing the automatic detection of discrepancies among these two sources. This is especially useful in cases where the switch is only partially opened, a condition not detected by standard telemetry instrumentation devices. Experiments with the system have been successfully done, where no action is taken if no error is detected. Otherwise, the system displays an alarm and gives the option to perform the 2D–3D registration, as shown in the next section.

5.3. User Interface Prototype

On startup, the VR client query the list of field cameras from the server. A red marker is added above the virtual instances of the monitored assets, indicating the initial state (no registration), as shown in Figure 7.

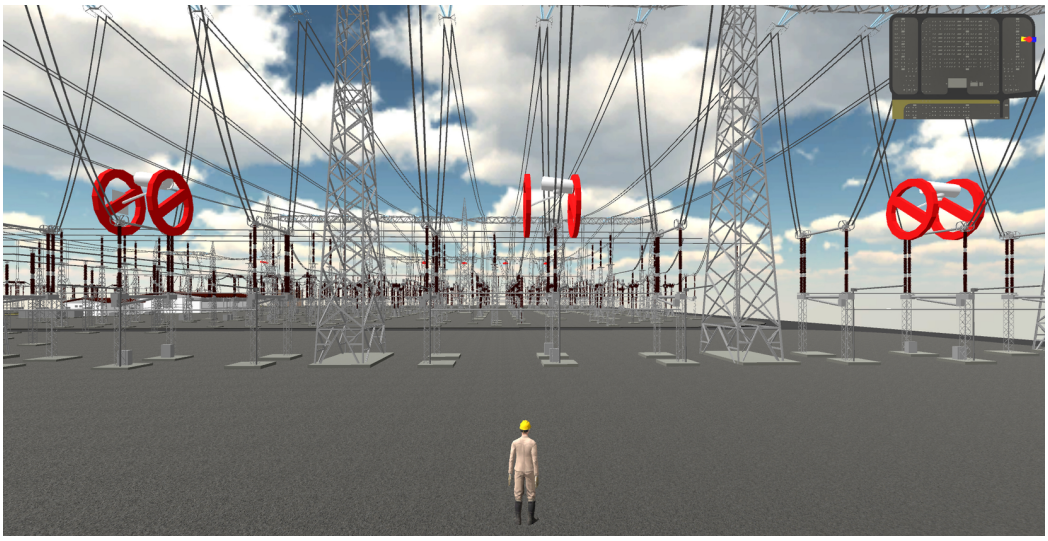


Figure 7. Initial state—all cameras disabled.

These markers are interactive, triggering a configuration dialog when clicked. The dialog (Figure 8) allows the selection of the camera (or none), shows some metadata obtained from the server and acts as an entry point for the focal length autoset (described in Section 4) and the 2D–3D registration.

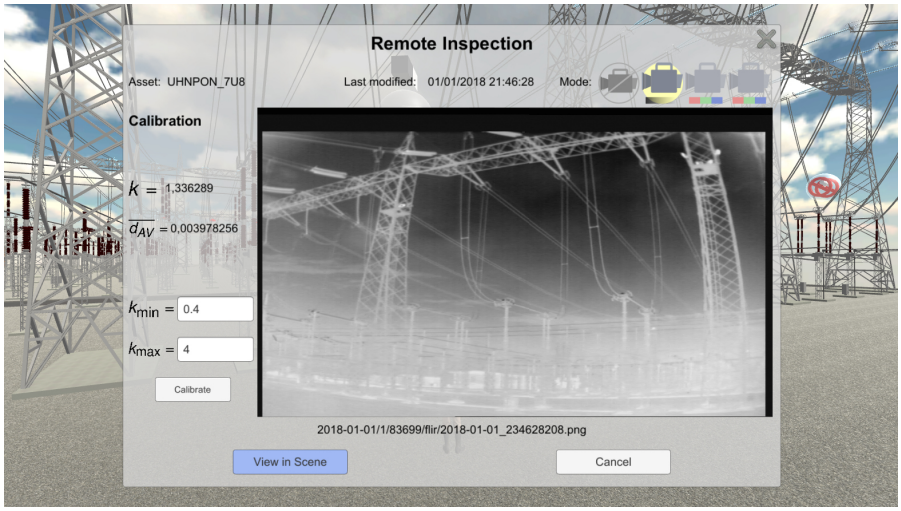


Figure 8. Field camera configuration dialog.

The client evaluates the registration quality in real-time (as explained in Section 6.1), disabling the overlay, whenever the error exceeds a threshold. In this scenario, the overlay is replaced by an icon indicating that condition. With this feature, navigation is not blocked in the augmented virtuality environment. Once the registration becomes poor, due to the high discrepancy between the virtual camera and the photo capture conditions, the overlay is simply disabled, avoiding misinterpretations from the operator.

Finally, if the system detects a discrepancy between the state reported by the SCADA database and the state inferred from the image, an alarm dialog is presented to the user (Figure 9).

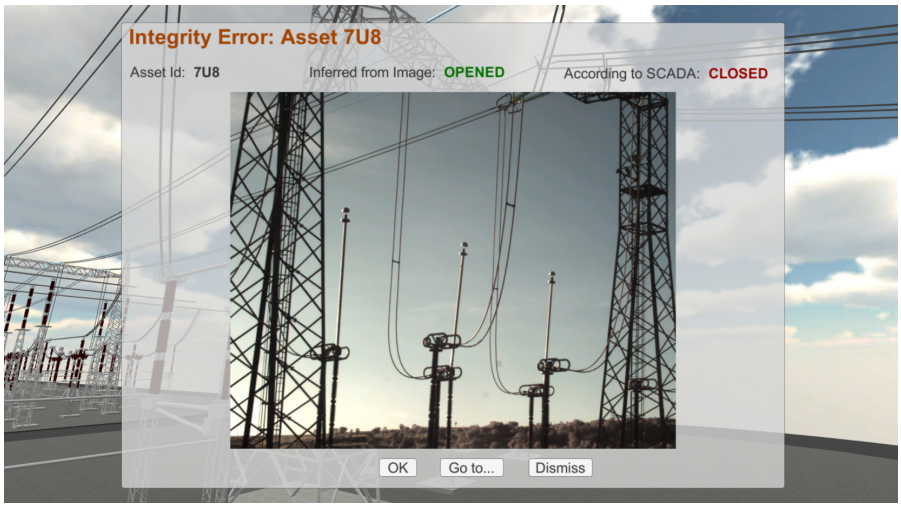


Figure 9. Alarm dialog.

The alarms can be either ignored (Dismiss button) or iconified (OK button). Alternatively, the dialog gives an option to teleport to the affected asset, for further inspection and contextualization. In this case, the environment's viewport is changed so that the photo is overlaid in the 3D model, leaving some space on the screen edges to see the surroundings, using the camera parameters stored in the database.

Multiple incoming alarms can be iconified, resulting in alarm queues for each inspected device. To avoid duplicates, the condition responsible for triggering each alarm is used for determining the alarm's lifecycle and identify. When this condition is no longer present, the system knows that a new, similar, discrepancy is supposed to result in a new alarm dialog.

6. Experimental Evaluation

The proposed method was developed and tested considering a virtual environment of a transmission substation (Figure 10), from a partner power company. This substation has a pair of cameras (RGB and Thermal) with PTZ control. They have five position adjustment presets, which, properly multiplexed in time, can be used to monitor three distinct power switches assets. The thermal camera always captures, in a single pose, the full geometry of the asset of interest. The color cameras, however, are adjusted to zoom levels requiring more than one pose to capture some of the assets.

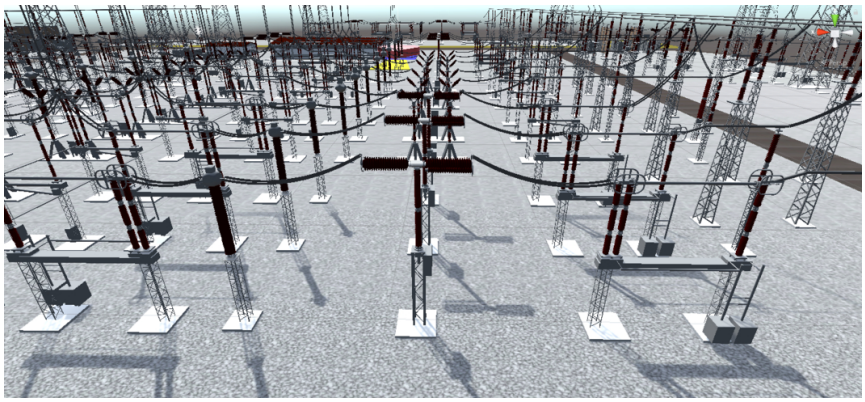


Figure 10. Power substation virtual environment.

The company kindly provided a dataset with 561 images captured by these cameras. From this set, 123 images were ignored since they correspond to images taken by RGB cameras without any favorable light conditions, at night. A sample of the dataset is presented in Figure 11.

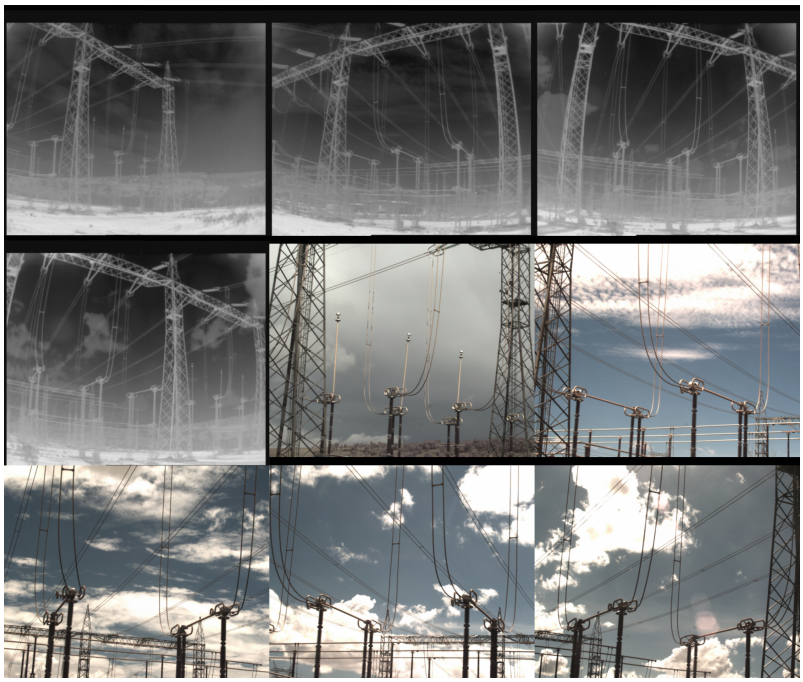


Figure 11. Thermal and RGB image dataset sample.

The images also have metadata for their timestamps, ranging from 31 December 2017 23:03 and 1 January 2018 22:46. This allows emulation of real-time data, by applying some time offset in the

system clock. The thermal camera images' size is 720 × 624 pixels, whereas RGB images' dimensions are 1280 × 1024.

Since the method uses points' correspondences, the prototype needs a convention for naming the keypoints, used both in the image and in world space. The convention used for this model is illustrated in Figure 12. Currently, no computer vision method was applied to detect these keypoints in the photos. They were manually specified for each camera's pose, using a single photo captured at that pose. Hence, mobile cameras or PTZ cameras with significant repositioning errors are not considered in this work.

Finally, the test used a simulated SCADA subsystem to arbitrarily set equipment states and thus allow triggering the alarm dialogs.

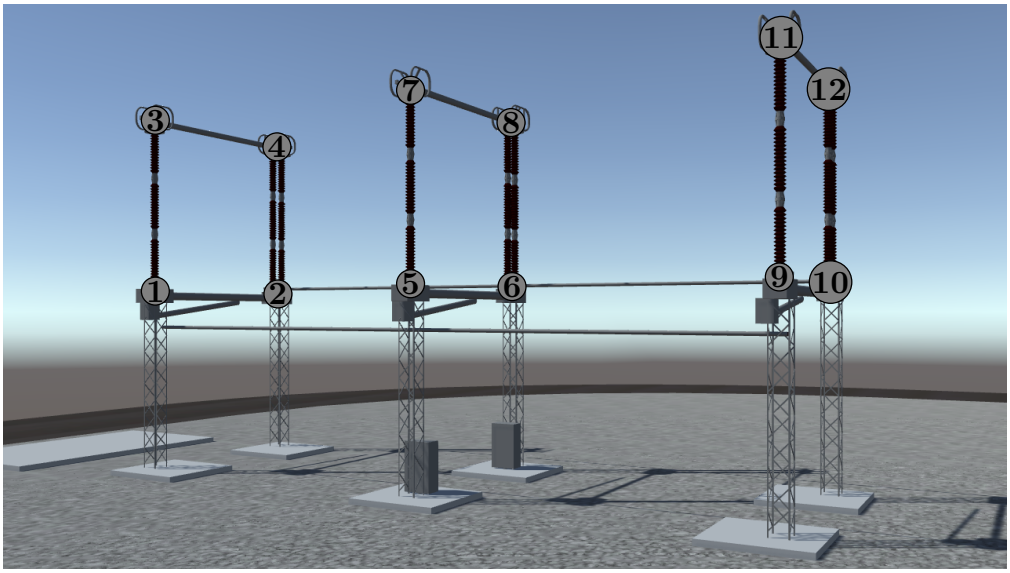


Figure 12. Power disconnector keypoints convention.

6.1. Registration Quality Metrics

The $\overline{d_{PNP}}$ metric described in Section 4 measures the mean distance in pixels, which might not be an intuitive unit for representing errors. A more generic alternative, then, is to obtain the relative errors for each axis, resulting in values that are independent of the image dimensions. The mean relative errors are given by:

$$\overline{e_{x\%}} = \frac{1}{n \cdot w_1} \sum_{i=1}^{n} \left| \eta \left(\boldsymbol{q}_i \right)_x - \eta \left(\rho_f \left(\boldsymbol{p}_i \right) \right)_x \right| \cdot 100\% \tag{17}$$

$$\overline{e_{y\%}} = \frac{1}{n \cdot h_1} \sum_{i=1}^{n} \left| \eta \left(\boldsymbol{q}_i \right)_y - \eta \left(\rho_f \left(\boldsymbol{p}_i \right) \right)_y \right| \cdot 100\% \tag{18}$$

where n is the number of keypoints, w_1 is the image width, h_1 is the image height, $\boldsymbol{q}_i$ is a vector in ICS whose coordinates of the pixel are related to the ith keypoint and $\rho_f \left(\boldsymbol{p}_i \right)$] is the result of the perspective projection of the WCS point located at $\boldsymbol{p}_i$, related to the ith keypoint, using the estimated camera pose.

Another metric consists of analyzing the keypoints positions once the virtual camera has been "teleported" to the estimated pose and the rectangular region has been textured with the field image. The Unity 3D scripting application programming interface (API) exposes a method to map a point in world space to the corresponding pixel related to the current camera viewport. This can be used to extract the keypoints' coordinates of the virtual model in the final rendered image.

For the rectangular region with the image overlay, invisible objects can be added as its children and positioned to match the photo keypoints. Once the plane is positioned and oriented, after the 2D–3D registration, the same world-to-screen utility method can be applied in these objects to extract their positions in the rendered image.

Again, the mean Euclidean distance can be used, considering the image rendered by the AV application. After 2D–3D registration succeeds, a screenshot is taken and the following parameter is calculated:

$$\overline{d_{AV}} = \frac{1}{n}\sum_{i=1}^{n}\left\|\eta\left(\boldsymbol{q}_{i,AV}\right) - \eta\left(\boldsymbol{q}_{i,M}\right)\right\|, \tag{19}$$

where n is the number of keypoints, $\boldsymbol{q}_{i,AV}$ is the ith key point pixel coordinates in the rectangular region and $\boldsymbol{q}_{i,M}$ is the ith key point pixel coordinates in the virtual model.

Relative errors for this metric are given below:

$$\overline{\delta_{x\%}} = \frac{1}{n \cdot w_1}\sum_{i=1}^{n}\left|\eta\left(\boldsymbol{q}_{i,AV}\right)_x - \eta\left(\boldsymbol{q}_{i,M}\right)_x\right| \cdot 100\% \tag{20}$$

$$\overline{\delta_{y\%}} = \frac{1}{n \cdot h_1}\sum_{i=1}^{n}\left|\eta\left(\boldsymbol{q}_{i,AV}\right)_y - \eta\left(\boldsymbol{q}_{i,M}\right)_y\right| \cdot 100\% \tag{21}$$

All three metrics $\overline{d_{AV}}$, $\overline{\delta_{x\%}}$ and $\overline{\delta_{y\%}}$ can be evaluated automatically by the VR application.

6.2. Performance Metrics

For measuring performance impact, one metric is related to the time elapsed between the instant just before a request, from the virtual environment, and the moment after which the server response has been processed.

For that matter, two kinds of requests are considered: (i) the calibration request, aimed at computing the optimal poses for the overlay plane and the virtual camera; and (ii) the image request, responsible for retrieving the last image from the database and updating the virtual environment accordingly. The symbols associated with these measurements are named Δt_{calib} and Δt_{photo}, respectively.

6.3. Results

Both calibration and registration were evaluated for nine different combinations of cameras and poses. The experiments were named using a two-character code. The first character is either 'T' or 'C', for thermal and color cameras, respectively. The second character is the index of the preset pose. Due to zoom levels, the RGB camera needs more poses for capturing the full geometry of some assets. Table 1 summarizes the conditions for each experiment.

Table 1. Experiments codes.

Code	Camera Type	Asset	Detail
T1	thermal	1	full
T2	thermal	2	full
T3	thermal	3	full
C1	color	1	lines A and B
C2	color	1	lines B and C
C3	color	2	lines A and B
C4	color	2	lines B and C
C5	color	3	full

All tests described in this section were performed on a Core i5-7400 CPU with 16 GB DDR4 RAM and no dedicated video card, running Windows 10 Home. Both the web server and the clients were deployed on the same physical machine.

The ternary-search algorithm was used to determine the optimal focal length scale factor. Figure 13 shows this behavior for 10 iterations, for the experiment T1 and the range $f_{x,y} \in [0.4, 4]$.

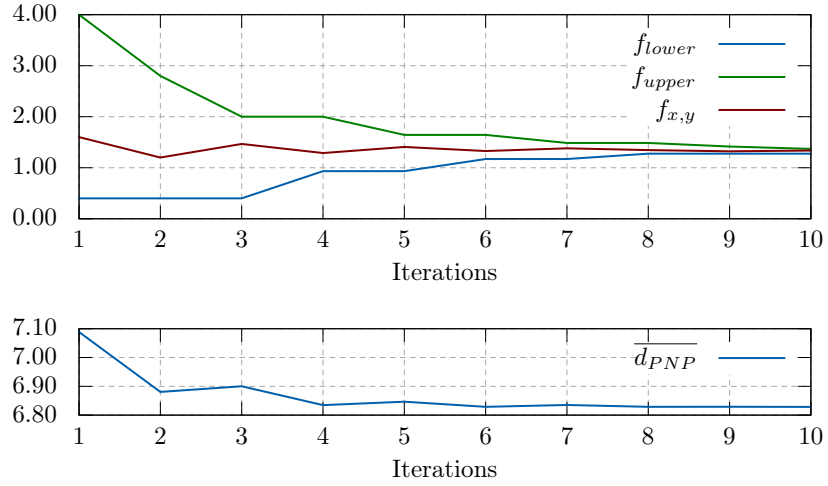


Figure 13. Ternary-search iterations.

The range for the search algorithm was determined empirically, from the inspection of the $\overline{d_{PNP}}$ values in a much broader range, as shown in Figure 14 for experiment T1.

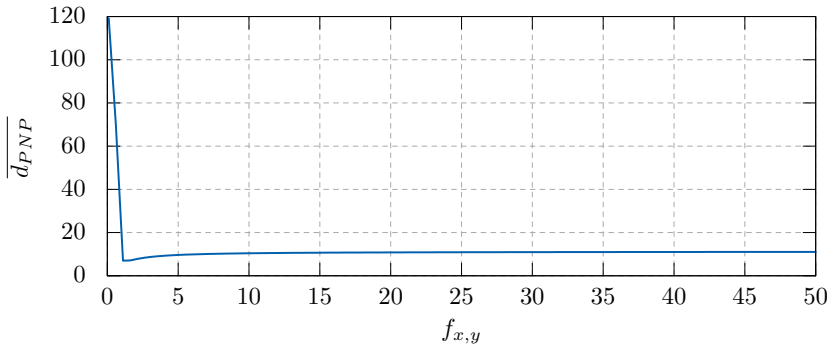


Figure 14. Focal length scale factor impact on $\overline{d_{PNP}}$ metric.

Data for other poses, along with some collected metrics, are summarized in Table 2. The algorithm was parameterized for running at most 50 iterations, also stopping in the nth iteration whenever $f_4 - f_1 < 10^{-5}$.

Table 2. *Perspective-n-Point* and calibration results.

Exp.	n	f	$\overline{d_{PNP}}$	$\overline{e_{x\%}}$	$\overline{e_{y\%}}$	Δt_{calib} (s)
T1	32	1.34	6.83	0.52%	0.80%	1.093
T2	32	0.94	6.97	0.59%	0.72%	1.004
T3	30	1.10	10.45	1.03%	1.01%	1.119
C1	32	1.30	2.13	0.11%	0.15%	1.079
C2	32	1.80	0.95	0.02%	0.08%	1.039
C3	30	2.00	8.05	0.60%	0.14%	1.057
C4	32	1.79	0.24	0.01%	0.02%	0.968
C5	32	2.13	1.80	0.07%	0.14%	1.030

Concerning the resulting rendered image after the registration, the keypoints' coordinates in pixels were extracted for both the rectangular region (overlay) and the virtual model instances. Figure 15 shows the resulting coordinates for experiments T1 and C1.

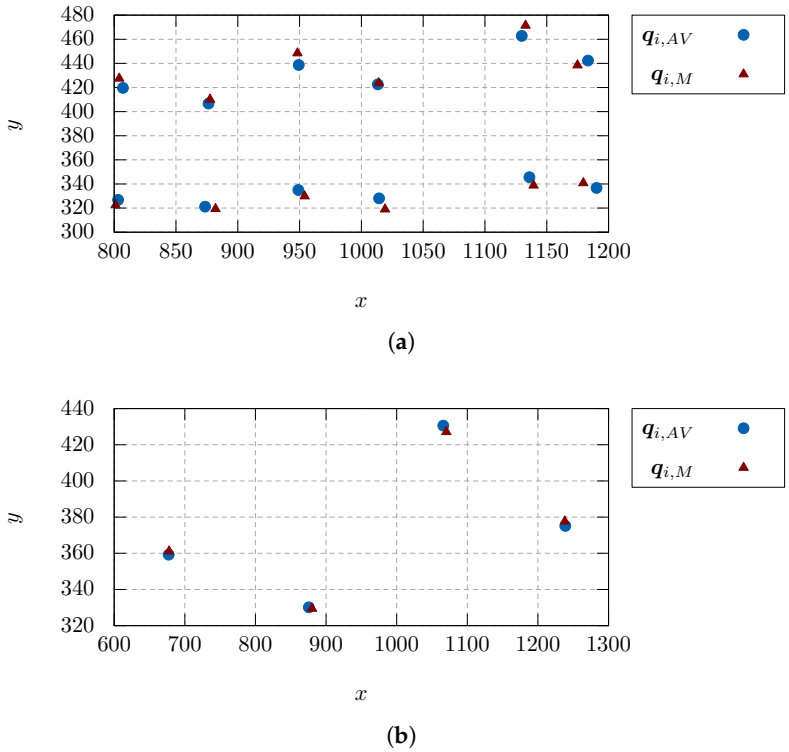


Figure 15. Keypoints discrepancy: (**a**) experiment T1; (**b**) experiment C1.

Table 3 gives the collected values for $\overline{d_{AV}}$, $\overline{\delta_{x\%}}$ and $\overline{\delta_{y\%}}$, for all experiments, as well as the average value for $\overline{\Delta t_{photo}}$, considering 20 requests.

The values obtained for $\overline{\Delta t_{photo}}$ are reasonable for real time remote inspection, especially for far locations with poor network bandwidth. It should be noted that the time needed to correctly interpret the situation, after the 2D–3D registration is performed, might be considerably longer than just a fraction of a second.

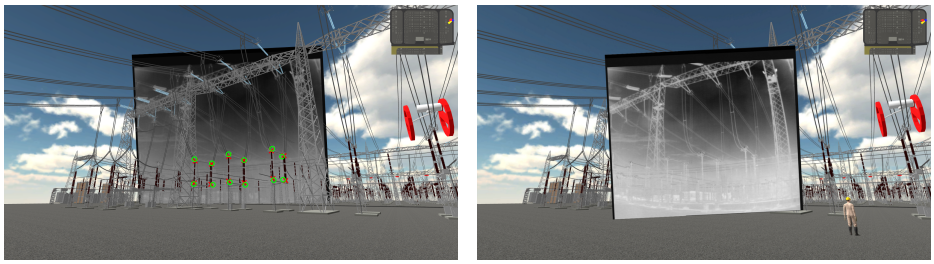
Table 3. Final registration results.

Exp.	$\overline{d_{AV}}$	$\overline{\delta_{x\%}}$	$\overline{\delta_{y\%}}$	$\overline{\Delta t_{photo}}$ (ms)
T1	47.59	0.226%	0.509%	380.111
T2	34.78	0.370%	0.657%	389.519
T3	31.91	0.254%	0.735%	382.404
C1	118.37	0.130%	0.200%	449.535
C2	165.56	0.045%	0.137%	440.236
C3	176.75	0.318%	0.143%	449.562
C4	164.06	0.050%	0.132%	441.255
C5	194.87	0.092%	0.153%	448.221

In addition, the power disconnector and porticos were modeled with incomplete computer-aided design (CAD) data, as opposed to more precise methods such as 3D scanning. This limitation directly affects the registration quality. Thus, the quantitative metrics are focused in the keypoints and the reprojection errors.

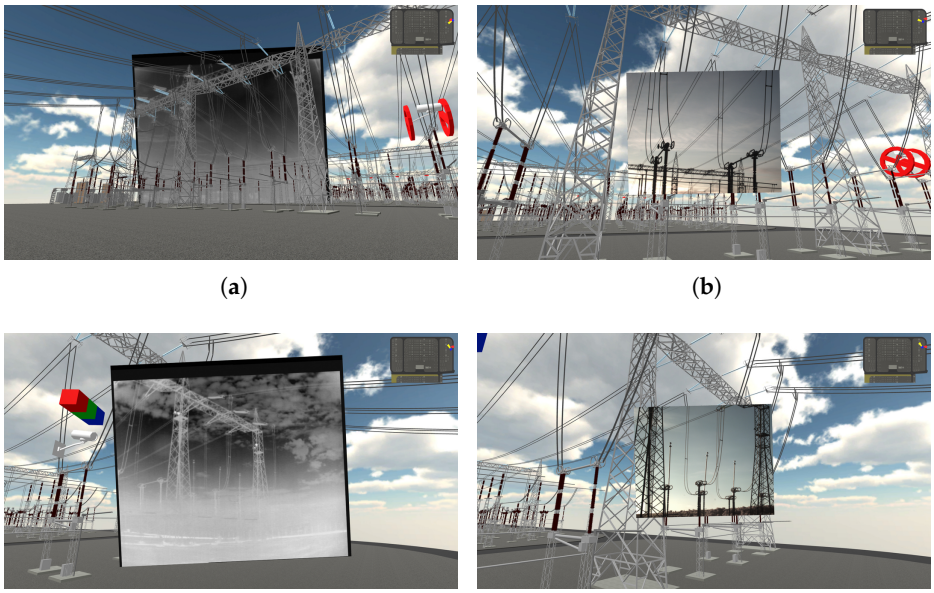
A custom shader was applied to the overlay plane to hide VR objects within its region. Figure 16 shows the rendered images for both standard and custom shaders. In the former, keypoints are highlighted

with red crosses (virtual model) and green circles (overlay rectangular region). Some other registrations are shown in Figure 17.



(**a**) (**b**)

Figure 16. 2D–3D registration for experiment T1: (**a**) custom shader and keypoints; (**b**) overlay.



(**a**) (**b**)

(**c**) (**d**)

Figure 17. Power disconnector registrations for other experiments: (**a**) T2; (**b**) C2; (**c**) T3; (**d**) C3.

7. Discussion

We evaluated the 2D–3D registration quality for the case of cameras without significant distortion and proposed a simple iterative algorithm to determine the focal length scale factor parameter so that keypoints correspondence is optimal. Other camera types, with significant skew factor, non-square pixels and other kinds of distortion, could be handled in future work either by submitting them to calibration methods [36] (p. 189) or by applying more complex optimization algorithms, with multiple parameters to be estimated.

However, considering a company responsible for dozens of power substations and having multiple inspection cameras with potentially different specifications, in-loco calibration is unpractical. To avoid expensive travels to many far locations, we propose using an approximated camera matrix and refining the focal length until an optimal value is found. In addition, with this approach, the deployment of the augmented virtuality environments can be done without shutting down the image inspection system or affecting the cameras' poses just for calibration purposes. Environments without any camera installed

remain VR-only and represent, in the proposed system's perspective, future candidates for deploying the RI system.

Once the optimal value for the focal length is determined, the PNP solver used in our system, from the OpenCV library [42], presented reasonable solutions (Figure 15 and Table 2), even with a small number of keypoints (just four in some poses for the RGB camera).

The overall matching depends on other factors, especially the virtual model's fidelity. In our virtual substation model, the porticos dimensions were not available as input data. However, the power disconnector model was based on some CAD drawings, so that the chosen keypoints act as a reliable ground-truth for evaluating the registration.

In this sense, the 2D–3D registration relative errors are arguably small, as shown in Figure 16a or summarized in Table 3, with the metrics $\overline{\delta_{x\%}}$ and $\overline{\delta_{y\%}}$. Since they are evaluated only after the rectangular region v_2 is positioned and oriented, following the mathematical model explained in Section 3, we can infer that the model has revealed itself appropriate for the problem at hand. This has some advantages, considering that "when the system model (or part of it) can be solved with analytical methods, considerable gains in terms of efficiency, accuracy, and understanding are usually obtained" [45].

A drawback of our approach is that the focal length scale factor range must be provided by the user during calibration. For experiments T3 and C3, which represent the pose capturing the furthest asset, the range had to be narrowed, to avoid solutions having the overlay plane too far away from the VR camera. Nonetheless, the calibration routine does not need to be run on every system startup, but only for configuring new fixed cameras. Considering the time needed to run the optimization, $\Delta t_{calib} \approx 1$ s, the process could be done in real-time, depending on the application. This is particularly useful for the scenario of mobile cameras if the keypoints' coordinates could also be extracted in real-time by computer vision techniques.

The client–server architecture has the benefit of caching the last images from each camera so that, if multiple clients are used in the operations center, fewer network requests are made to the image database. In addition, since the focal length optimization is done occasionally and on the server-side, VR clients can spend their processing resources on more important tasks, notably real-time rendering. However, the system's architecture performance remains to be tested as future work.

The proposed user interface combines interactive virtual objects placed near the monitored assets, 2D–3D spatial registration whenever the virtual camera's pose is adequate, and customizable camera settings for each device. Although currently tested only in one substation, it is already prepared for multiple cameras and regions of interest.

Concerning the SCADA integration, faults can be better understood by allowing the user: (i) to be notified whenever there is an inconsistency, as shown in Figure 9; and (ii) to immediately see the last field image and, if desired, the 2D–3D spatial image registration using that image. This is already implemented, but some user experiments are still required to provide a comprehensive evaluation of the feature.

Registration took approximately 384 ms for the thermal images and 486 ms for RGB images, including the HTTP request and response times and the VR rendering. The slight difference is most likely due to the image sizes: 720×624 for the former against 1280×1024 for the latter. Taking into account the trends on 5G mobile networks, this can be an issue once real-time video registration is needed. An alternative would be to use the server only for the PNP solving and calibration, and to open a dedicated User Datagram Protocol (UDP) video channel between the client and the substation, querying images and updating the plane texture accordingly. For the RI of power substations, having the field image updated two times per second seems adequate for nearly real-time operation.

Finally, since the registration metric $\overline{d_{AV}}$ is computed in real-time by the client, it is possible to tolerate small variations on the estimated virtual camera's pose until the error exceeds a threshold. This feature is already implemented in the system.

Further work consists of deploying the solution into an operations center and evaluating the operators' performance in the electrical system reestablishment, using power flow simulators. Additionally, the system could be adapted and tested in other environments needing similar RI or teleoperation facilities, such as construction machines [46], marine systems [19], or industrial boilers.

Author Contributions: Conceptualization, L.M., A.C. and E.L.; methodology, L.M.; software, L.M.; validation, L.M.; formal analysis, L.M. and E.L.; investigation, L.M.; resources, A.C.; data curation, L.M.; writing—original draft preparation, L.M.; writing—review and editing, L.M., A.C. and E.L.; visualization, L.M.; supervision, A.C.; project administration, A.C.; and funding acquisition, A.C. and E.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Minas Gerais Energy Company (CEMIG) and the Brazilian Electricity Regulatory Agency (ANEEL), through project GT-0618. This study was also financed in part by Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Brasil (CAPES), Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) and Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cardoso, A.; Lamounier, E.; Lima, G.; do Prado, P.; Ferreira, J.N. VRCEMIG: A Novel Approach to Power Substation Control. In Proceedings of the ACM SIGGRAPH 2016 Posters, Association for Computing Machinery, Anaheim, CA, USA, 24–28 July 2016; pp. 3:1–3:2. [CrossRef]
2. Pal, D.; Meyur, R.; Menon, S.; Reddy, M.J.B.; Mohanta, D.K. Real-time condition monitoring of substation equipment using thermal cameras. *IET Gener. Transm. Distrib.* **2018**, *12*, 895–902. [CrossRef]
3. Hongkai, C.; Zhao, X.; Tan, M.; Sun, S. Computer vision-based detection and state recognition for disconnecting switch in Substation automation. *Int. J. Robot. Autom.* **2017**, *32*, 1–12. [CrossRef]
4. Sebe, I.O.; Hu, J.; You, S.; Neumann, U. 3D Video Surveillance with Augmented Virtual Environments. In Proceedings of the First ACM SIGMM International Workshop on Video Surveillance (IWVS'03), Association for Computing Machinery, New York, NY, USA, 7 November 2003; pp. 107–112. [CrossRef]
5. Vaidya, S.; Ambad, P.; Bhosle, S. Industry 4.0—A Glimpse. *Procedia Manuf.* **2018**, *20*, 233–238. [CrossRef]
6. Álvaro Segura.; Diez, H.V.; nigo Barandiaran, I.; Arbelaiz, A.; Álvarez, H.; Simoes, B.S.; Posada, J.; García-Alonso, A.; Ugarte, R. Visual computing technologies to support the Operator 4.0. *Comput. Ind. Eng.* **2020**, *139*, 105550. [CrossRef]
7. Longo, F.; Nicoletti, L.; Padovano, A. Smart operators in industry 4.0: A human-centered approach to enhance operators' capabilities and competencies within the new smart factory context. *Comput. Ind. Eng.* **2017**, *113*, 144–159. [CrossRef]
8. Postolka, B.; List, R.; Thelen, B.; Schütz, P.; Taylor, W.R.; Zheng, G. Evaluation of an intensity-based algorithm for 2D/3D registration of natural knee videofluoroscopy data. *Med. Eng. Phys.* **2020**, *77*, 107–113. [CrossRef] [PubMed]
9. ISO. *Information Technology—Computer Graphics, Image Processing and Environmental Data Representation—Mixed and Augmented Reality (MAR) Reference Model*; Standard ISO/IEC 18039:2019; International Organization for Standardization: Geneva, Switzerland, 2019.
10. Vagvolgyi, B.; Niu, W.; Chen, Z.; Wilkening, P.; Kazanzides, P. Augmented virtuality for model-based teleoperation. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 3826–3833. [CrossRef]
11. Meng, C.; Wang, Q.; Guan, S.; Sun, K.; Liu, B. 2D–3D registration with weighted local mutual information in vascular interventions. *IEEE Access* **2019**, *7*, 162629–162638. [CrossRef]
12. Yoshiya, S. Utilization of image analysis in joint surgery. In Proceedings of the 2017 6th International Conference on Informatics, Electronics and Vision 2017 7th International Symposium in Computational Medical and Health Technology (ICIEV-ISCMHT), Himeji, Japan, 1–3 September 2017; p. 1. [CrossRef]
13. Karakottas, A.; Papachristou, A.; Doumanoqlou, A.; Zioulis, N.; Zarpalas, D.; Daras, P. Augmented VR. In Proceedings of the 2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), Reutlingen, Germany, 18–22 March 2018; p. 1. [CrossRef]
14. Lv, L.; Li, S.; Wang, H.; Jin, L. An approach for fault monitoring of insulators based on image tracking. In Proceedings of the 2017 24th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), Auckland, New Zealand, 21–23 November 2017; pp. 1–6. [CrossRef]

15. Pereira, R.F.L.E.C.; Vieira, E.; Moreira, L.F.E.; Tamietti, M.V.G.; Conselho, T.H.B.; Silva, A.S.F. Sistema para Videomonitoramento Operacional de Chaves Seccionadoras de Subestações de Energia [A System for Operational Videomonitoring of Power Substations Disconnector Switches]. In Proceedings of the 14th Encontro para Debates de Assuntos de Operação (XIV EDAO), São Paulo, Brazil, 21–23 November 2016.
16. Hu, Y.; Wu, W.; Zhou, Z. Video Driven Pedestrian Visualization with Characteristic Appearances. In Proceedings of the 21st ACM Symposium on Virtual Reality Software and Technology (VRST'15), Beijing, China, 1–4 November 2015; pp. 183–186. [CrossRef]
17. Wu, Y.; Liu, C.; Lan, S.; Yang, M. Real-Time 3D Road Scene Based on Virtual-Real Fusion Method. *IEEE Sens. J.* **2015**, *15*, 750–756. [CrossRef]
18. Xie, J.; Zhou, Y.; Wu, W.; Zhou, Z. Automatic Path Planning for Augmented Virtual Environment. In Proceedings of the 2016 International Conference on Virtual Reality and Visualization (ICVRV), Hangzhou, China, 24–26 September 2016; pp. 372–379. [CrossRef]
19. Nađ, D.; Mišković, N.; Omerdic, E. Multi-Modal Supervision Interface Concept for Marine Systems. In Proceedings of the OCEANS 2019—Marseille, Marseille, France, 17–20 June 2019; pp. 1–5. [CrossRef]
20. Kim, C.; Lee, J.; Han, T.; Kim, Y.M. A hybrid framework combining background subtraction and deep neural networks for rapid person detection. *J. Big Data* **2018**, *5*, 1. [CrossRef]
21. Liu, Y.; Wu, W.; Wu, Z.; Zhou, Z. Fire Detection in Radiant Energy Domain for Video Surveillance. In Proceedings of the 2015 International Conference on Virtual Reality and Visualization (ICVRV), Beijing, China, 4–5 November 2015; pp. 1–8. [CrossRef]
22. Shi, L.; Long, F.; Lin, C.; Zhao, Y. Video-Based Fire Detection with Saliency Detection and Convolutional Neural Networks. *Advances in Neural Networks—ISNN 2017*; Cong, F., Leung, A., Wei, Q., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 299–309. [CrossRef]
23. Wang, T.; An, Q.; Li, J.; Zhang, Y.; Han, J.; Wang, S.; Sun, S.; Zhao, X. Vision-based illegal human ladder climbing action recognition in substation. In Proceedings of the 2017 Ninth International Conference on Advanced Computational Intelligence (ICACI), Doha, Qatar, 4–6 February 2017; pp. 189–194. [CrossRef]
24. Changfu, X.; Bin, B.; Fengbo, T. Research of Substation Equipment Abnormity Identification Based on Image Processing. In Proceedings of the 2017 International Conference on Smart Grid and Electrical Automation (ICSGEA), Changsha, China, 27–28 May 2017; pp. 411–415. [CrossRef]
25. Sevak, J.S.; Kapadia, A.D.; Chavda, J.B.; Shah, A.; Rahevar, M. Survey on semantic image segmentation techniques. In Proceedings of the 2017 International Conference on Intelligent Sustainable Systems (ICISS), Palladam, India, 7–8 December 2017; pp. 306–313. [CrossRef]
26. Li, H.; Zhang, Y.; Liang, G. Application of Foreground Detection Technology in Intelligent Video Monitoring System of Substation. In Proceedings of the 2Nd International Conference on Computer Science and Application Engineering, CSAE '18, Hohhot, China, 22–24 October 2018; pp. 111:1–111:5. [CrossRef]
27. Luo, Y.; Tu, G. Who's watching the unattended substation [substation television system]. *IEEE Power Energy Mag.* **2005**, *3*, 59–66. [CrossRef]
28. Xiaoming, S.; Shaosheng, F.; Bing, Y. Implementation of infrared measuring temperature on remote image monitoring and control system in transformer substation. In Proceedings of the 2012 International Conference on Image Analysis and Signal Processing, Vienna, Austria, 11–13 April 2012; pp. 1–4. [CrossRef]
29. Buhagiar, T.; Cayuela, J.P.; Procopiou, A.; Richards, S. Poste intelligent—The next generation smart substation for the French power grid. In Proceedings of the 13th International Conference on Development in Power System Protection 2016 (DPSP), Edinburgh, UK, 7–10 March 2016; pp. 1–4. [CrossRef]
30. Antonijević, M.; Sučić, S.; Keserica, H. Augmented Reality Applications for Substation Management by Utilizing Standards-Compliant SCADA Communication. *Energies* **2018**, *11*, 599. [CrossRef]
31. Xiao-le, H.; Xiao-bo, W.; Fang-dong, C.; Xu, H.; Xue-min, F.; Lin, L. 500 kV substation robot patrol system. In Proceedings of the 2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 3–5 October 2017; pp. 105–109. [CrossRef]

32. Cai, D.; Huang, Q.; Li, J.; Chang, Z. A practical preset position calibration technique for unattended smart substation security improvement. In Proceedings of the 2017 IEEE Power Energy Society General Meeting, Chicago, IL, USA, 12–16 July 2017; pp. 1–5. [CrossRef]
33. Kaehler, A.; Bradski, G. *Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library*, 1st ed.; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2016.
34. Vince, J. *Mathematics for Computer Graphics*; Springer: London, UK, 2006.
35. Unity Technologies. Unity. 2020. Available online: https://unity.com/ (accessed on 29 September 2020).
36. Hartley, R. *Multiple View Geometry in Computer Vision*; Cambridge University Press: Cambridge, UK; New York, NY, USA, 2004.
37. Fischler, M.A.; Bolles, R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* **1981**, *24*, 381–395. [CrossRef]
38. César, V.M.; Farias, T.; Macedo, S.; Kelner, J.; Santos, I. Avaliação de Algoritmos de Estimativa de Pose para Reconstrução 3D e Realidade Aumentada [Evaluation of Pose Estimation Algorithms for 3D Reconstruction and Augmented Reality]. In Proceedings of the VIII Workshop de Realidade Virtual e Aumentada (WRVA 2011), Uberaba, Brazil, 7–9 November 2011.
39. Lepetit, V.; Moreno-Noguer, F.; Fua, P. EPnP: An Accurate O(n) Solution to the PnP Problem. *Int. J. Comput. Vis.* **2009**, *81*, 155–166. [CrossRef]
40. Dementhon, D.F.; Davis, L.S. Model-based Object Pose in 25 Lines of Code. *Int. J. Comput. Vis.* **1995**, *15*, 123–141. [CrossRef]
41. Brown, M.; Windridge, D.; Guillemaut, J.Y. A family of globally optimal branch-and-bound algorithms for 2D–3D correspondence-free registration. *Pattern Recognit.* **2019**, *93*, 36–54. [CrossRef]
42. Bradski, G. The OpenCV Library. *Dobb J. Softw. Tools* **2000**. 25, 120–125.
43. Grinberg, M. *Flask Web Development: Developing Web Applications with Python*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2018.
44. Hipp, D.R.; Kennedy, D.; Mistachkin, J. SQLite. 2020. Available online: https://www.sqlite.org/ (accessed on 21 September 2020).
45. Heiliö, M.; Lähivaara, T.; Laitinen, E.; Mantere, T.; Merikoski, J.; Pohjolainen, S.; Raivio, K.; Silvennoinen, R.; Suutala, A.; Tarvainen, T.; et al. *Mathematical Modelling*; Springer International Publishing: Berlin/Heidelberg, Germany, 2016; doi:10.1007/978-3-319-27836-0. [CrossRef]
46. Iwataki, S.; Fujii, H.; Moro, A.; Yamashita, A.; Asama, H.; Yoshinada, H. Visualization of the surrounding environment and operational part in a 3DCG model for the teleoperation of construction machines. In Proceedings of the 2015 IEEE/SICE International Symposium on System Integration (SII), Nagoya, Japan, 11–13 December 2015; pp. 81–87. [CrossRef]