

Research Article

ipProjector: A Vision-Based Framework for Unsupervised Ubiquitous Object Augmentation Using a Portable Projector-Camera System

Thitirat Siriborvornratanakul and Masanori Sugimoto

Interaction Technology Laboratory, Department of Electrical Engineering and Information Systems, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

Correspondence should be addressed to Thitirat Siriborvornratanakul, thitirat@itl.t.u-tokyo.ac.jp

Received 13 July 2010; Revised 25 November 2010; Accepted 9 February 2011

Academic Editor: Karen Panetta

Copyright © 2011 T. Siriborvornratanakul and M. Sugimoto. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ipProjector stands for interactive portable projector which is a project trying to make a smart projection system while retaining absolute portability of the devices and setups. In this paper, an ipProjector framework is proposed for unsupervised real-time object augmentation applications. Scene-independent projector-camera geometric calibration is introduced so that the projected augmentation always appears in the desired geometry despite an unconstrained surface. Using nonintrusive projection, the system can ensure correct vision-based analysis because external appearances of the environment and object (as seen by the camera) are hardly changed by the overlaid projected content. Finally, an unsupervised visual detector is used to guarantee robustness of the detection regardless of lighting variations. Several image processing algorithms are applied in the framework, including image warping using perspective transformation, color conversion using the nearest-neighbor search algorithm, and multiscale visual detection. Based on a small-scale laboratory setting, experiments were conducted to evaluate the proposed approaches, and one example was built to prove the proposed concept in object augmentation.

1. Introduction

Because of the trend in mobile projector phones, people have started to imagine using mobile projectors ubiquitously in the same way as mobile phones. A good ubiquitous system is a matter of mobility and intelligence. While the mobility of a projector has recently been successfully proposed, intelligent mobile projection has not yet been accomplished in practice. In this paper, we focus on a vision-based ubiquitous projection system that is able to perform real-time object augmentation in a robust manner. Only geometry aspect is concerned, and our goal is to create a system that can project 2-D augmentations at the desired position corresponding to the target object in a 3-D environment.

Object augmentation using a projector has been proposed continuously in the past decade. This trend has become more prominent with the recent proposal of the SixthSense project [1] that uses a wearable projector to

augment the physical world with virtual information projected from a projector. The vision-based projector-camera (i.e., pro-cam) system is often the choice for researchers in this field because it allows object augmentation everywhere without relying on separate tracking hardware, and a user can interact directly with the projection.

There are two main problems encountered when making a pro-cam system aware of its surrounding objects, while still retaining ubiquity. First, ubiquitous projection implies use in an unknown environment that has not been prepared. However, geometrically calibrating the pro-cam system requires creating a strong geometric relationship between projector, camera, and surface. Hence, the unconstrained surface becomes an important factor that makes the calibration become very difficult. The second problem, which we call “projective interference”, is caused by the projected contents being captured by the camera. Suppose that the camera is working in the visible spectrum at a standard shutter speed,

the projected contents overlaid on the environment are also seen by the camera. For vision-based systems, this leads to false detections of the environment and object unless the system has a way to deal with the interference effectively.

Apart from the two mentioned problems, there is no doubt that the key success of object augmentation is to detect the target object correctly in the first place. This means that a visual object detector used in the system must be robust enough in a real-world scenario. However, there is a fundamental conflict using projection and visual detection together in the visible light spectrum. Namely, projection requires a dark environment for better visualization of the projected images on the surface whereas the visual object detection algorithm usually prefers input images captured in a bright environment so that the sharp details of objects are obtained with few additive noises. Combined with the fact that visual appearances of the object can be changed easily by the environment lightings, the problem of visual object detection becomes more challenging for a ubiquitous pro-cam system.

All proposed approaches are designed to be robust to unknown geometries, textures, and variations of lighting in a real-world scenario of ubiquitous projection. Contributions of this paper can be applied to other vision-based pro-cam systems to ensure robust geometry-based object augmentation whose accuracies are barely affected by environment lightings or appearances of the surface and object. Flexibility is another advantage of this framework. Any off-the-shelf digital light processing (DLP) projector can be used to implement a system following this framework. A representative detector used in this paper can also be replaced by another visual object detector with no internal modification to the detector. Finally, this framework supports integration of other vision algorithms (e.g., tracking algorithm, gesture recognition, adaptive learning, etc.), enabling realistic object augmentation to achieve in an actual application.

The rest of this paper is laid out as follows. Section 2 explains recent advances in object augmentation using a projector and then discusses related research in the three problem areas as mentioned earlier. Section 3 introduces the proposed framework and the overall system configuration. Section 4 presents a self-contained and portable pro-cam design whose geometric calibration can be achieved on both planar and nonplanar surfaces with no prior knowledge of their geometry. Section 5 describes problems of variation of lighting regarding the proposed design and presents our solution of the unsupervised visual object detection. Section 6 then explains the nonintrusive projection technique using a DLP projector to solve the projective interference problem. A simple program is built and shown in Section 7 to demonstrate the overall concept of the proposed framework. In Section 8, performances of the proposed approaches are experimentally confirmed. Finally, Section 9 concludes this article.

2. Related Work

As mentioned, our achievement is related to three types of problems. In the following sections, we explain recent

advances in object augmentation using a projector and then discuss related research in the three areas.

2.1. Object Augmentation by Projection. So far, researchers in the field of ubiquitous pro-cam systems focus on developing new interactive techniques but pay scant attention to the fundamental problems (as mentioned in Section 1) in a real-world scenario. Previous works of augmenting objects by projection tend to neglect or simplify these problems with various alternatives that do not meet the goal of being ubiquitous. Paper-based fiducials are attached to the target object in SixthSense [1], iLamps [2], and Shelf Torchlight [3]. The rectangular handheld display screen is recognized and tracked by its black border in [4] and by four light sensors attached to its corners in [5]. Three infrared LEDs forming a triangular shape are placed on the robot allowing robot manipulation using a mobile projector in CoGAME [6]. So far, it can be said that using a visual marker is the most popular solution for a ubiquitous pro-cam system. It significantly simplifies the problems of a ubiquitous pro-cam system, enabling creation of a system that can extract only the desired information and disregard the others (including information suffering from the projective interference). Nevertheless, these works are not ideally ubiquitous yet, because they either modify or engineer external appearances of the object or surface beforehand.

Other works that do not incorporate with alternation of object appearances are shown in [7–9]. The work of Kanbara et al. [7] uses a projector to project invisible markers which can only be seen by the specific camera. An environment-aware display system proposed by [8] embeds the imperceptible stripe pattern into the normal projection, and real-time object detection is achieved without any marker. A commercial 3-D tracking unit is used in [9], enabling physical object annotation to be performed based on the 3-D position tracking strategy. All these works can solve the early problem of using visual markers by not modifying external appearances of the object (as seen by users). However, none of them retains absolute ubiquity and portability in their implementation [7] requires the invisible markers to be projected steadily onto a wall or ceiling, and moving the projector or object is not allowed. The system in [8] is limited to fixed projectors and fixed cameras mounted on a ceiling, and projection surfaces are restricted to flat table surfaces whose distance to the ceiling is unchanged. In [9], one stationary camera is needed in the workspace for assisting the 3-D tracking unit. The cooperative augmentation proposed by [10] seems to be the appropriate solution for real-time object augmentation in ubiquitous projection. Their pro-cam system dynamically configures its visual object detection based on four different detection algorithms that ensure detection coverage in a real-world scenario. Nonetheless, this system relies on an assumption of the smart object where object-model knowledge must be embedded during manufacture.

Unlike these systems, our proposed framework and design allow creating a markerless system that is truly self-contained and well calibrated despite the unconstrained environment and dynamic objects. Besides, the framework

is automatic and involves no user feedback, training, or supervision during online execution.

2.2. Pro-Cam Geometric Calibration. So far, researchers have tried proposing a new approach to calibrating the pro-cam system precisely on different surfaces. When a projector and a camera are rigidly fixed to each other, some have assumed that the geometric registration between them is roughly constant [4]. However, as the angle of the projector moves from the perpendicular, or as a surface becomes nonplanar, this approach will no longer guarantee good geometric registration. In Map TorchLight [11], a mobile phone is fixed to a mobile projector, and a known paper-based map (containing visible black dots arranged in a regular grid) is used to assist real-time geometric calibration. Projecting a known pattern onto a surface is a classical approach to solve this problem that gives precise calibrations for both planar surfaces [12–14] and nonplanar surfaces [2, 15, 16]. However, the computational cost is high for a complex surface, and the pattern must be reprojected when any component of the system (e.g., a projector, camera, or surface) moves. A similar approach is applied in the catadioptric projectors [17] whose projected light is refracted and reflected by refractors and reflectors, respectively. Geometric registration between the two devices is obtained by projecting a series of known patterns and allowing the camera to sense them. A real-time approach that does not interrupt normal projection was proposed in [18] by attaching four laser pens to a handheld pro-cam device. Although detecting bright laser points sounds easy, locating small laser points in a complex camera image is still difficult in practice. In [19], Johnson and Fuchs proposed a real-time approach that does not interrupt the normal projection, requires no fixed marker, and can be applied to a complex surface. By matching feature points found in the projected image and the predicted captured image, the pose of the projector is tracked and the calibration is achieved in real time. However, the camera is stationary and separated from the projector in their system.

Unlike these systems, our proposed calibration is truly ubiquitous. It can perform the calibration in real time without interrupting the normal projection or requiring any stationary device in the workspace. Besides, both planar and nonplanar surface can be used as the projection surface with no prior knowledge of their geometry.

2.3. Nonintrusive Projection. Nonintrusive projection can be seen as an inverse problem of the embedded imperceptible pattern projection whose projected pattern is invisible to users and normal cameras but visible to the calibrated camera. There are many solutions proposed for the imperceptible pattern projection but few for the nonintrusive projection. Fortunately, some solutions can be shared between these two problems with slight modifications. In the office of the future [20], structured light can be embedded into a DLP projector by making significant changes to the projection hardware. However, this implementation is impossible unless it is incorporated into the design of the projector or full access to the projection hardware is available. In [21–23], a

code image is projected at high speed with its neutralized image which integrates the coded patterns invisibly due to limitations of the human visual system. According to these papers, projecting and capturing at 120 Hz can guarantee a hidden code, but commonly available projectors usually perform projections at a maximum rate of 87 Hz. These previous works try to invisibly embed the binary pattern to the normal projection. By changing the embedded pattern to the uniform white image (the reason regarding this will be explained at the end of Section 6.2), some of them can perform nonintrusive projection.

For this paper, we apply an approach based on the DLP characteristics. Recently, internal characteristics of a DLP projector have received attentions from research communities. In [24, 25], two approaches utilizing the internal characteristics of a DLP projector are proposed. First, the dithered illumination pattern corresponding to the DMD chip (which operates at 10,000 Hz) is observed and utilized using a very high-speed camera (whose maximum speed is 3000 fps). Second, characteristics of the color wheel (which rotate at 120 Hz) is investigated and used by a camera with slower capturing speed. Our nonintrusive projection is based on the color wheel solution and has three main advantages: it requires no internal change to the projector or the camera, it can be applied to any off-the-shelf DLP projector, and it supports embedded variable light patterns (i.e., imperceptible pattern projection) in the future without further hardware modifications.

2.4. Robust Visual Object Detection. Variations in lighting are the classic problems that have been discussed in computer vision communities for decades, particularly for surveillance, cognitive, or mobile systems where a specific environment cannot be assumed. Recent researches of robust object detection have focused on applying visual learning or statistical modeling so that a specific object or a specific class of objects can be detected in all possible scenarios. For model-based approaches, many works try to build a generic model of the target object applicable for all lighting scenarios. For example, face detection under variant illumination video sequences is solved in [26] by building a skin-tone model and segmenting each video frame into skin and nonskin regions. The idea of the model-based approach is to train the system using an exemplar set representing appearances of the target object under all scenarios. Two drawbacks of this approach are that the exemplar set may not be available or adequate for some applications, and this approach does not support detecting an unknown object. Instead of building a model of the target object, researches in [27, 28] reduce sensitivity to variations of lighting by modeling the background and then using background subtraction to extract the foreground (including the target object). Although these systems are adaptive to variations of lighting and allow detection of an unknown object, they assume uses in the specific background only. A similar idea is applied in [29] for a cognitive mobile robot system. Assume that the robot acts in a closed environment and repeatedly returns to the same place, the statistical model of lighting behaviors is built, enabling the robot to predict lighting of the environment at a

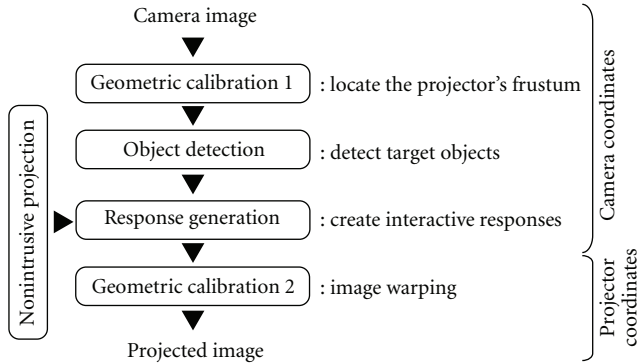


FIGURE 1: Procedural flow of the proposed framework during the online execution.

specific location. Using a model limits the system to perform detection only where the model is available as discussed. To solve this problem, a robust adaptive object detection that does not rely on a model created offline is shown in [30]. By using an online learning algorithm, the system can adapt to changing environments with no need for the predefined background model. Nevertheless, this system also requires environment-specific information to reduce the number of false alarms. Therefore, it cannot be applied in a portable system.

It can be seen that detecting an unknown object in variations of lighting is still a challenging problem in computer vision. Fortunately, our proposed system is barely affected by the unpredictable environment lighting because the projector's red light is the only light source illuminating the environment as seen by the camera (as detailed in Section 6). Therefore, variations of lighting can be simply explained as an inverse relationship of the distance from the projector to the object (please refer to Figure 5(a)). In other words, variations of lighting in our system are not as complex as the related works in this field. Hence, we decided to apply the multiscale concept to ensure detection coverage in our system. Other reasons that we chose this multiscale concept are as follows: (1) it requires no internal modification in the base detector, (2) it is flexible because the base detector can be replaced easily with another visual detector to match individual requirements, and (3) increases in computation can be easily offset by the recent growth of multicore processors and parallel programming languages.

In this paper, we propose a complete vision-based framework and a portable pro-cam design for ubiquitous object augmentation applications. The goal is to accomplish a pro-cam system that (1) retains absolute ubiquity despite the projection environment, (2) allows projecting real-time augmentation everywhere, and (3) is able to reliably detect objects inside the projector's frustum without supervision. The framework consists of three main parts as the solutions for the two mentioned problems and the robust visual object detection. Experiments were conducted to evaluate the proposed approaches in a small-scale laboratory setting. Actual object augmentation application has not been implemented

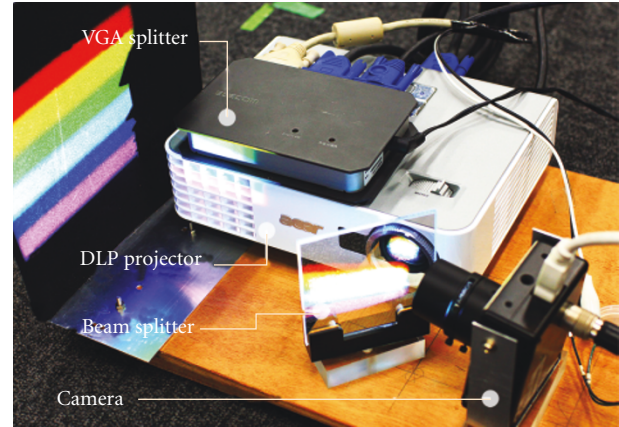


FIGURE 2: Overall system configuration.

in this paper, and only one program projecting simple augmentation is shown as the proof of concept.

3. Framework Overview and System Configuration

In this section, we briefly explain the proposed framework and the system configuration used in the paper. As illustrated in Figure 1, our proposed framework for real-time object augmentation consists of five main tasks (written inside the rounded rectangles) during the online execution. First, the projection frustum appearing inside a camera image is located (*Geometric calibration 1*). Then, the target object existing inside the frustum area is detected (*Object detection*). After that, interactive augmentation is calculated for each detected object (*Response generation*). A *nonintrusive projection* is applied in this step to guarantee that all augmentations are drawn using the correct colors. Up to this point, all calculations are done in camera coordinates. Finally, another calibration is performed to convert every projection content to projector coordinates (*Geometric calibration 2*) so that it appears in our desired geometry on the actual surface; this task is sometimes referred to as image warping.

Information regarding *geometric calibration 1* and *2*, *object detection*, and *nonintrusive projection* are shown in Sections 4, 5, and 6, respectively. The *response generation* is performed using different algorithms (depending on the application), so a detailed explanation of it is not considered here. Following this framework, the system is able to detect the target object and project the corresponding augmentation in our desired geometry. Note that this framework focuses only on the fundamental tasks based on the frame-by-frame analysis strategy. Additional algorithms can be added before the *response generation* step, enabling accurate detection or realistic augmentation in a real-world application. For example, adding a tracking algorithm will help a system to augment with the target object(s) in a continuous manner, using gesture recognition will support direct user interaction with the projected image, and applying object recognition will allow recognizing a specific

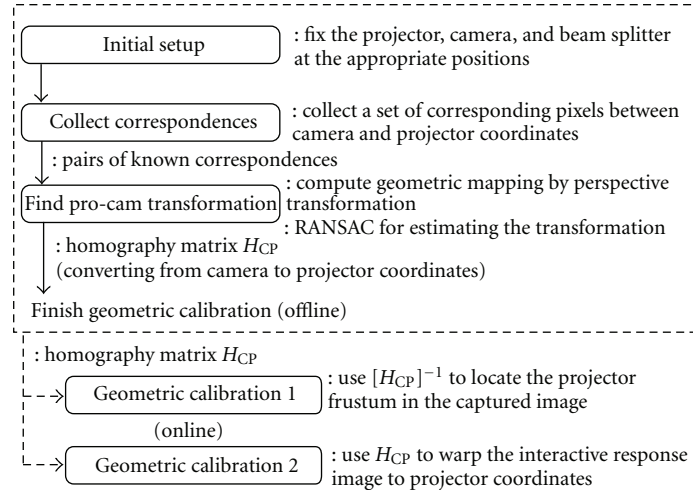


FIGURE 3: Procedural flow of the proposed scene-independent pro-cam geometric calibration during offline and online execution.

object or a specific class of objects whose appearances match requirements.

Our system configuration is shown in Figure 2. A camera, beam splitter and DLP projector are fixed firmly on a wooden base forming a single self-contained device (see Section 4 for details) whereas a VGA splitter is added for pro-cam synchronization (see Section 6 for details). The device is completely self-contained and portable; however, re-engineering is required to make it more compact for ubiquitous uses.

4. Scene-Independent Pro-Cam Geometric Calibration

Any pro-cam system is related to three pairs of geometric calibration: projector-camera, projector-surface, and camera-surface calibration. For an application such as the automatic keystone correction (using a vision-based solution), all three calibrations have to be performed concurrently. However, for projecting 2-D augmentations at the desired location regardless of the 3-D shapes of the surface, only pro-cam calibration is adequate. Geometric mapping between camera and projector coordinates is used to find corresponding positions between the two coordinate systems and to project images back to desired locations on an actual surface. These two tasks refer to the *geometric calibration 1* and 2, respectively, as illustrated in Figure 1.

Pro-cam geometric calibration is one of the problems of most concern in a ubiquitous projection system. Because, although the projector and the camera are fixed together, poses and orientations of the surface (compared to the two devices) are still unknown. On a planar surface, when the angle of a projector moves away from perpendicular, the geometric mapping changes. On a nonplanar surface, there is an additional problem as 3-D shapes of the surface create parallax effects between projector and camera coordinates. This nonoverlapping fields of view between the projector and the camera may make geometric mapping impossible

in some positions. Using a 3-D scanner to render the 3-D model of the surface is an ideal solution that tracks relative positions and orientations of the projector compared to the surface in real time. However, this is still difficult to perform accurately in real time using a portable device in an unknown environment, particularly when the reference device (i.e., projector and camera) and the target object are not stationary.

Figure 3 shows the summary of our calibration approach. The offline processes (shown inside the dashed-line rectangle) consist of setting up the beam splitter and computing geometric mapping between projector and camera coordinates. The outcome of these offline processes is the homography matrix representing the geometric transformation from camera coordinates to projector coordinates and vice versa. During the online execution, there is no need to recalculate this matrix, and the geometric calibration can be achieved regardless of 3-D geometry of the projection surface.

In the following sections, we explain in detail how to achieve the mentioned geometric calibration using a single self-contained device. Section 4.1 (which refers to the *initial setup* step in Figure 3) shows a collocating pro-cam design whose geometric calibration is considered to be independent of the surface. Section 4.2 (which refers to the rest of the offline processes and the online processes written in Figure 3) then describes how to use perspective transformation model to calculate pro-cam geometric mapping based on the design presented in Section 4.1.

4.1. Collocating the Projector and Camera. Previous researches on mobile projection have assumed projections on a uniform planar surface, which is the most primitive projection surface. The reason behind this assumption is to eliminate any effects due to the surface geometry. In previous researches, although a camera was firmly fixed to a mobile projector, their optical axes were not exactly aligned. Therefore, geometric conversions between them were affected by the geometry of the projection surface. Using 3-D surface rendering is an indirect solution of this problem; however,

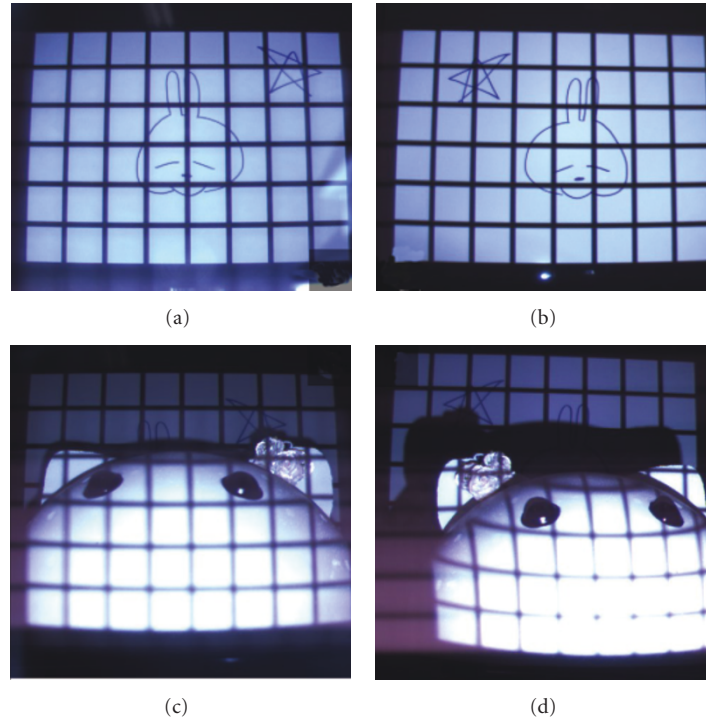


FIGURE 4: Comparison between the noncolocated (b, d) and colocated (a, c) pro-cam devices. (a, b) show comparisons on a planar surface. (c, d) show comparisons on a curved surface.

it is still difficult to perform in real time for a mobile system. To achieve a robust pro-cam system, we use a straightforward solution by collocating the projector and camera so that the surface factor is eliminated from the calibration process. As a result, geometric conversions between the two coordinates hardly change and can be said as being independent of the surface.

A beam splitter is an optical device that reflects half of the incoming light and transmits the other half. There are few researches concerning a beam splitter for pro-cam systems. Our design proposed in this section was inspired by the work of Fujii et al. [31] which briefly described the idea of scene-independent geometric calibration using a plate beam splitter attached to an off-the-shelf projector. Both their research and ours operate in the visible light spectrum; however, the camera settings are completely different. In their research, a camera uses a standard shutter speed and works independently to a projector. In our research (as described later in Section 6), the camera is accurately synchronized with the DLP projector and its shutter is opened for only 0.55 ms.

By collocating the two devices using a beam splitter (as shown in Figure 2), the camera sees exactly what a projector is projecting, and 3-D geometry of the surface hardly affects the geometric conversions or causes parallax between projector and camera coordinates. An example is shown in Figure 4. There is parallax between the two coordinates when using a noncolocated pro-cam device on a nonplanar surface (Figure 4(d)) so that some parts of the projected surface cannot be seen by the camera. Compared

with Figure 4(c) of the collocating design, the shape of the surface does not cause significant distortion of the projected pattern and shadowing is also reduced.

In summary, this design is very useful for geometric calibration that requires high conversion accuracy between the two coordinates. An example is shown in [24] where the collocating design is used for direct-global separation in order to accurately map camera pixels to the corresponding projected intensities. For ubiquitous projection, this design is very interesting because (1) it requires no external device stationary in the workspace, (2) precise geometric calibration is possible on any surface, (3) no additional computations are required, (4) it ensures that any surface visible to the camera can be projected upon, and (5) it eliminates shadows in camera images. Nevertheless, this approach is not suitable for pro-cam applications that utilize distortions of the projected images on the surface. For example, as occurs with 3-D reconstruction using projected structured light.

4.2. Perspective Transformation Model. In this paper, geometric mapping between camera and projector coordinates is computed by perspective transformation [32] whose computation effort is lighter than Euclidean calculation, suitable for real-time interactive applications. Based on the fact that all points seen by the camera lay on some unknown plane, the perspective transformation between the two coordinates can be established by a 3×3 homography matrix. Suppose that (X, Y) is a pixel in projector coordinates whose corresponding pixel in camera coordinates is (x, y) , the perspective transformation from (x, y) to (X, Y) can be

expressed with eight degrees of freedom in homogeneous coordinates as

$$\begin{pmatrix} Xw \\ Yw \\ w \end{pmatrix} = \begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \quad (1)$$

where $\vec{H}_{CP} = (h_1 \dots h_9)^T$ is constrained by condition $|\vec{H}_{CP}| = 1$ and can be computed from four corresponding pixels between the two coordinates (four correspondences ensure that no three points is collinear). When there are more than four corresponding pixels found between the two coordinates ($\delta > 4$ in (2)), the RANSAC method is applied for estimating the values of \vec{H}_{CP} in the following equation:

$$\begin{pmatrix} X_1w & X_2w & \dots & X_\delta w \\ Y_1w & Y_2w & \dots & Y_\delta w \\ w & w & \dots & w \end{pmatrix} = \begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{pmatrix} \cdot \begin{pmatrix} x_1 & x_2 & \dots & x_\delta \\ y_1 & y_2 & \dots & y_\delta \\ 1 & 1 & \dots & 1 \end{pmatrix}. \quad (2)$$

During the offline processes, the system collects at least four corresponding pixels between the two coordinates (e.g., by projecting a known pattern on the surface) and uses them to compute \vec{H}_{CP} values following (2). After that, geometric mapping from any (x, y) to (X, Y) and vice versa is achieved using \vec{H}_{CP} and $[\vec{H}_{CP}]^{-1}$, respectively. Combined with the colocalizing pro-cam design (presented in Section 4.1) whose geometric transformation is not affected by the geometry of the surface, recomputation of this homography matrix is not necessary if there is no change in the relative positions or orientations among the projector, camera, and beam splitter. In this way, scene-independent geometric calibration is accomplished while ubiquity of the device is retained.

5. Multiscale Visual Object Detection

Because of the colocalizing design shown in Section 4 and the short-exposure setting as described later in Section 6, the fundamental lighting difficulties of our vision-based system are significantly underlined. Available light as seen by the camera is very limited, and the projector's red light is the only light source illuminating the projection surface. Small changes in distance from the projector to the object or slight depth variation of the surface can result in considerably increasing or decreasing the amount of light as seen by the camera.

Figure 5 shows luminance of the projection as seen by the camera in three setups, namely, Setup 1 for direct camera capture with no beam splitter, Setup 2 for direct camera capture with the beam splitter in front of the projector, and Setup 3 for our pro-cam setup using indirect camera capture via the beam splitter. Using the camera setting as written in Section 6.1, we constantly projected a uniform red image onto a whiteboard located at different distances and allowed the camera to sense these red projections in the three setups. According to the result shown in Figure 5(a), it is clear that the amount of light seen by the camera in our setup is

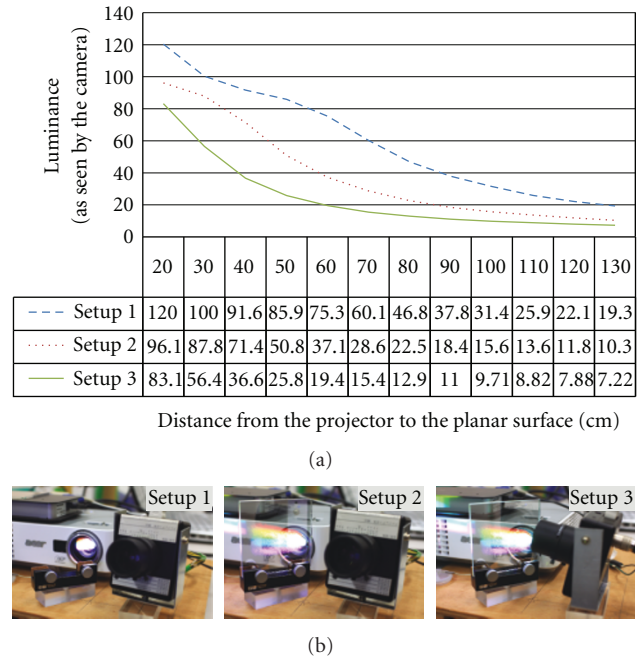


FIGURE 5: (a) Luminance values seen by the camera at different distances to the surface in three different setups. Note that the maximum possible luminance here is 255. (b) The three setups used in the experiment.

very limited and sharply decreased with the increase of the distance. Note that the distance written in this context refers to a distance from the planar surface to the front edge of the wooden base (as shown in Figure 2).

This significant decrease of luminance makes a visual object detection become difficult for a conventional standalone detection algorithm, as the detector tends to miss some true objects (a.k.a. undersegmentation) or include false information belonging to noises (a.k.a. oversegmentation). Considering a single input image, it is always possible to achieve a good detection result by fine-tuning parameters of the detector. However, adjusting these parameters manually during real-time object augmentation is not preferred in our system. Therefore, we decided to use the multiscale concept to compensate the oversegmented result with the undersegmented result and to ensure that acceptable detection results will always be achieved with a single set of predefined parameters, regardless of different or limited lighting conditions.

In the following sections, we explain in detail how to develop our proposed multiscale detection. Section 5.1 explains the overall concept of the proposed detection scheme. Section 5.2 presents additional information regarding appropriate image simplification filters for the proposed detection and the proposed pro-cam design.

Unlike color and texture, edges generally present across any type of visual content. Hence, Canny edge detection provided by the OpenCV library [33] is used in the following sections as a representative base detector where, λ_1 , λ_2 , and σ represent values of two Canny thresholds and aperture size,

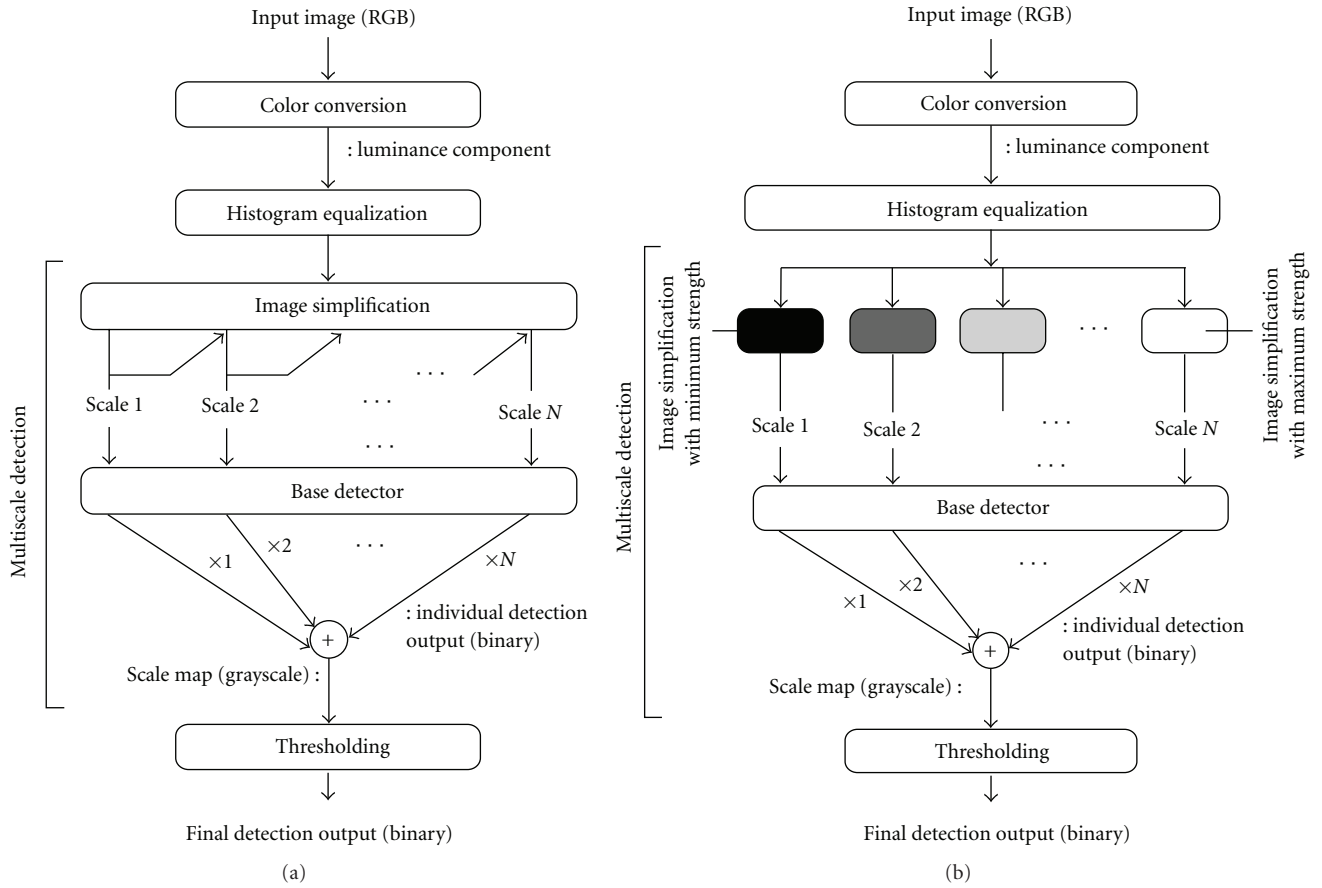


FIGURE 6: Procedural flows of the proposed multiscale visual detection scheme using sequential (a) and parallel (b) implementation.

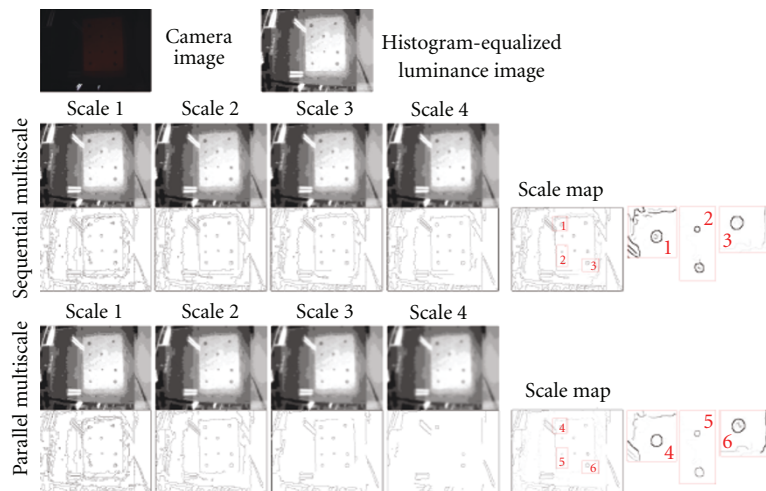


FIGURE 7: Comparison between the sequential and parallel multiscale detection. Gaussian smooth filter and Canny edge detector are used as the simplification filter and base detector, respectively.

respectively. Note that all input images used in these sections are unsigned 8-bit RGB images. We strongly recommend the reader to look at the digital copy of these sections because some fine details cannot be seen when printed onto a paper.

5.1. Overall Concept. An overview of the investigated detection scheme is shown in Figure 6. It starts by converting the image to YUV format and using only the Y (luminance) component in further calculations. This is related

to the observation that human eyes are far more sensitive to luminance than color. Then, histogram equalization is applied to enhance visibility and to standardize the input image so that it is ready for the next multiscale detection step. From our experiments, using only the multiscale concept (explained in the next paragraph) is not enough to achieve an unsupervised detection and adjusting parameters of the base detector is still required when dealing with differently illuminated input images. Because our input images (as well as other pro-cam systems operating in a dark environment) suffer from insufficient amounts of light, using histogram equalization is a straightforward solution to improve the overall appearance of the images, as shown in Figure 7.

Generally, the multiscale detection refers to a hierarchy of detections where an original image is simplified through iterations of a simplification filter as illustrated in Figure 6(a). The idea is to use an image simplification filter (with constant parameters) to decrease iteratively “too detailed” information from the original image so that only the information belonging to the true objects survives in the last scale. By applying the base detector with the constant set of parameters to the simplified image in each scale, we can combine the detection result from all scales to a single scale-map output using a weighted summation. Suppose that N is the number of scales and i is an index of each scale, in this paper, i equals to 1 refers to the highest scale (maximum detail but least importance), i equals to N refers to the lowest scale (minimum detail but highest importance), and the weight assigned to each scale is equal to i . Consequently, the high magnitude (black) in the combined scale map reflects the high probability that this pixel belongs to the true edges, as shown in Figure 7.

Nevertheless, using the sequential implementation means that the total number of scales is the key factor to determine the computational time of the detection. Considering recent growth in multicore processors and parallel programming languages, we changed the sequential implementation to the equivalent parallel implementation as illustrated in Figure 6(b). The histogram-equalized image is distributed to all scales simultaneously, allowing multiple simplifications to be performed in a parallel manner. Instead of using the same simplification parameters iteratively as the sequential implementation, our parallel implementation increases the simplification effect by directly enlarging the parameters that control the strength of simplification in each scale. The experimental results presented in Figure 7 show that the parallel implementation offers similar scale-map outcomes compared with the sequential implementation. In this way, speed of the detection can be improved significantly with few modifications.

After the multiscale detection step, the scale map image (grayscale) is converted to a binary image using a fixed threshold value so that low-magnitude pixels, which usually belong to noises, are eliminated from the final detection result. The investigated detection scheme is automatic and involves no user feedback, training, or supervision during online execution. By simply replacing the representative detector used in this section (Canny edge detector) with

a preferred visual detector, various object detections can be achieved for further use in object augmentation. The proposed detection scheme can also be applied to other pro-cam systems regardless of our colocating and nonintrusive design. However, some parameters may need to be reconsidered offline to ensure that the simplification strength and the detector’s sensitivity are set appropriately.

5.2. Image Simplification Filter. In addition to the detection scheme mentioned in Section 5.1, there is an additional issue of increasing textural noises caused by the histogram equalization. The textural noise in this context refers to textures that are hardly recognized in an original image but become noticeable when strongly enhancing the image with the histogram equalization. The more the histogram is changed, the clearer textural noises become and can result in false positive visual detection. Therefore, using an appropriate filter to simplify the image is important to ensure that these textural noises are not emphasized in the detection output. For this section, we experimented on five smooth filters: mean shift, conventional bilateral, real-time $O(1)$ bilateral [34], median, and Gaussian filters. Meanshift and bilateral filters are edge-preserving smooth filters that offer “posterized” or “cartoonized” effects over an input image. Median and Gaussian filters are well-known smooth filters usually used to reduce noises in an image.

Assuming that parameters of each filter are adjusted so that the best detection result is achieved, Figure 8 shows the scale maps obtained by applying the five filters over the same histogram-equalized image in the parallel multiscale manner. The number of scales (N) is 4, and the parameters of the base Canny edge detector are set to $\lambda_1 = 127$, $\lambda_2 = 1000$ and $\sigma = 5$ in all experiments. From Figure 8, it is obvious that the meanshift and median smooth filters emphasize the textural noises in the scale maps whereas the conventional bilateral and Gaussian smooth filters offer the cleanest and nicest edges. Compared with the conventional bilateral filter, the real-time $O(1)$ bilateral filter [34] can equally filter out the textural noises, but does better in preserving edges and enhancing image contrast (the result regarding this issue is not shown in this paper). However, its results are distracted by noise pixels inside the object so that continuous contours of an object are barely achieved. Note that only the area inside the projector frustum is considered here.

Furthermore, we examined the computational time used by each filter. Under the same constraints, the computation time used per one smoothing execution (without iteration) is 30, 63, 65, 18, and 15 ms for meanshift, conventional bilateral, real-time $O(1)$ bilateral [34], median, and Gaussian filters, respectively. Because our input image here requires strong smoothing effects to reduce the textural noises caused by the histogram equalization, it costs much time in the simplification processes. When dealing with other scenarios of less severe light conditions (not too dark), reducing strength of the simplification filter is allowed and less computation is required. Nevertheless, considering the time as well as the resulting scale map, the Gaussian smooth filter is the best choice that is not too sensitive to textural

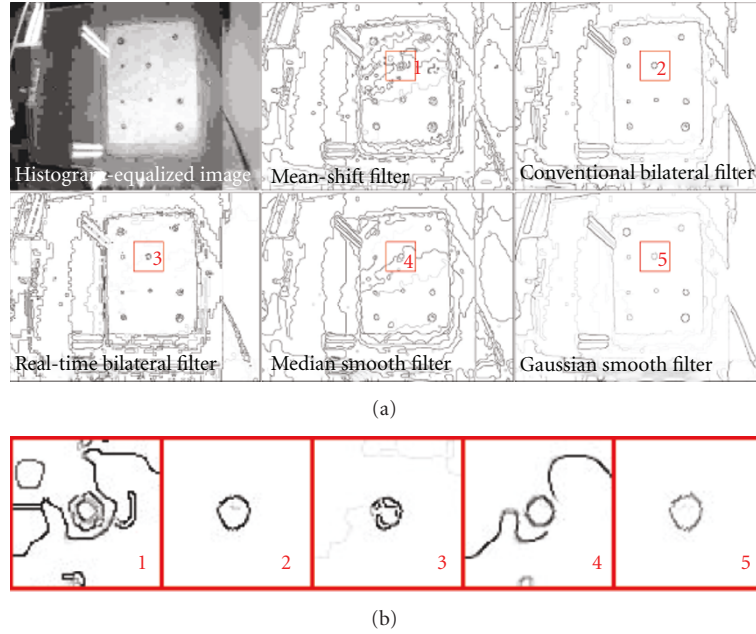


FIGURE 8: The scale maps of the multiscale Canny edge detection ($N = 4$) using different simplification filters. Top left is the histogram-equalized image before the multiscale detection is applied.

noises, provides nice contours of the true objects, and uses reasonable computational times.

6. Nonintrusive Projection

In order to achieve smart ubiquitous projection, the projection environment must be aware and recognized in real time so that projected augmentations can be generated in the most appropriate manner. Real-time environment analysis is not new in research communities but, it is not straightforward for a real-time projection system. In vision-based systems, it is still difficult to differentiate projected contents from actual objects in camera images, particularly when the projection surface is not planar and uniformly colored. This problem is hardly mentioned in the work of previous researchers investigating mobile projectors. Previously, the focus was on the projection functions or the use of markers to easily extract environmental information (as discussed earlier in Section 2). In the electronic sticky notes and projected augmented reality approaches proposed in [18], environment analysis is performed to detect an object in the environment whose appearances match the preregistered values. The interesting point here is that they seem to avoid projecting anything over the target object because projected contents superimposed on the target object will affect the visual matching process. In [3, 35], projecting over the target object is allowed but the projected information is reduced to a minimum to avoid significant changes in the object appearances as seen by the camera.

Generally, there are two alternatives to deal with projective interference so that environment analysis and projection can be performed simultaneously. One alternative creates a system that can be aware of all projected content at all

times. Handling projected contents in camera images is not so difficult if they do not overlap with physical objects. The SixthSense project [1] utilizes this concept and keeps tracking the projected content so that the system can handle it correctly. This alternative certainly requires additional computation for tracking and handling with the projected content. However, as the projection surface contains many details or textures, tracking the overlaid projected content and handling with it should become much more difficult. The other alternative is to prevent the projected content from being seen by the camera in the first place. The straightforward approach to achieve this is to observe the environment using a sensing device insensitive to light (e.g., depth camera, laser scanner). For this paper, we selected the second alternative in order not to risk system inefficiencies due to the surface complexity. The light sensing camera is used but in a specific setting, allowing the projected content to be invisible to the camera. Our approach is called “nonintrusive projection” which refers to the projection approach whose projected contents are visible to humans and normal cameras but invisible to the synchronized camera, as an example shown in Figure 9(E). Therefore, the vision-based system can avoid projective interference (as seen by the camera) that may lead to an incorrect environment analysis like that shown in Figure 9(C).

Figure 10 shows the summary of our nonintrusive projection approach. The offline processes (shown inside the dashed-line rectangle) consist of synchronizing the projector and the camera, analyzing the color wheel characteristics of a DLP projector, and classifying the RGB color space into three categories. The outcomes of these offline processes are the specific camera settings and three sets of classified colors. During the online execution, only the color conversion is

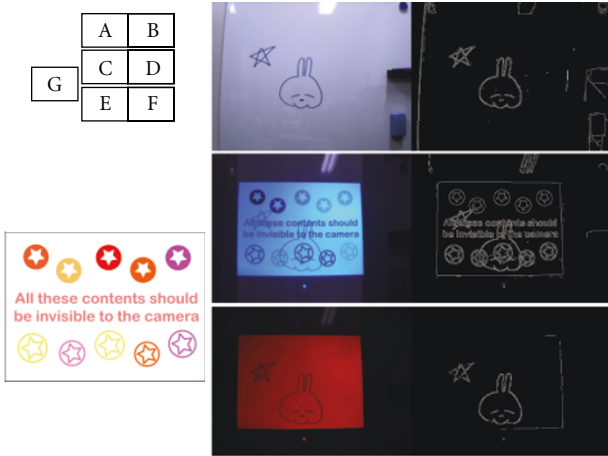


FIGURE 9: (A) is an environment seen by the normal camera. (C) and (E) show environment (A) as seen by the normal camera and the synchronized camera (according to the nonintrusive projection), respectively, while (G) is being projected from the projector. (B), (D), and (F) are Canny edge detection results of (A), (C), and (E), respectively.

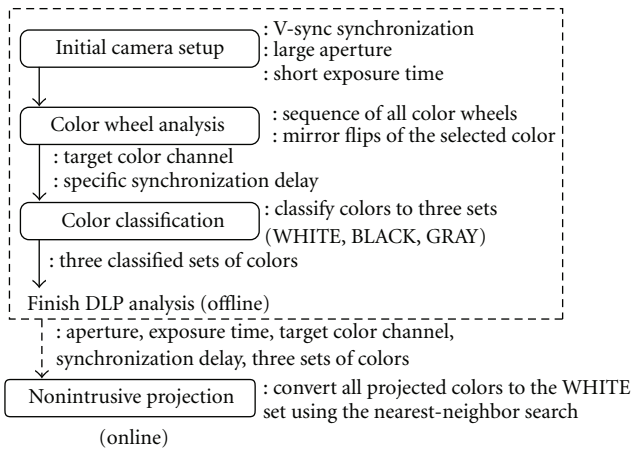


FIGURE 10: Procedural flow of the proposed nonintrusive projection during offline and online execution.

performed to ensure projecting only colors in the set whose visibility cannot be sensed by the synchronized camera.

Section 6.1 shows in detail how to perform the first two steps of the offline processes (as written in Figure 10) which mainly involve DLP color wheel analysis. Then, Section 6.2 explains the utilization of Section 6.1 for nonintrusive projection; this section refers to the remaining step of the offline processes and the online processes, as shown in Figure 10. Note that a beam splitter (as described in Section 4) has not yet been applied in these sections.

6.1. Color Wheel Analysis of a DLP Projector. In this section, characteristics of the color wheels are analyzed for further uses in Section 6.2. Because each DLP projector model owns unique characteristics of the inside color wheel, color wheel analysis has to be performed before using an unknown DLP

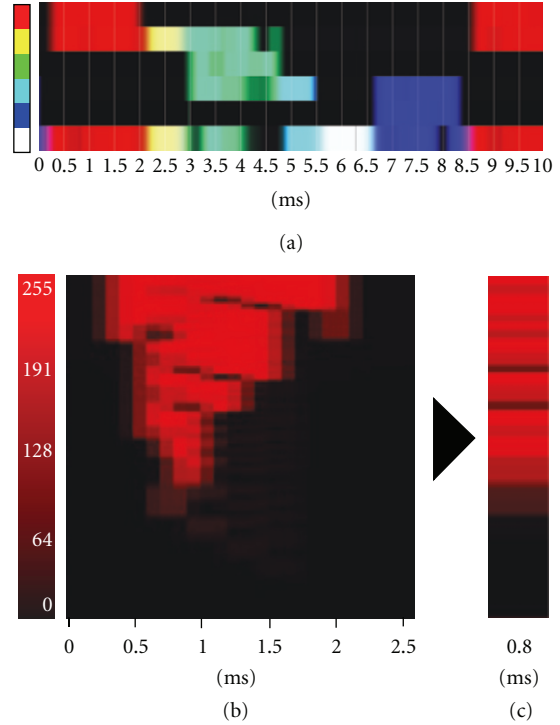


FIGURE 11: Color wheel analysis of the Acer P3250 DLP projector. (a) is the overall color-wheel sequences as seen by the synchronized camera with starting exposure times ranging from 0 to 10 ms. (b) is detailed mirror flip sequences for all 256 values in the red channel with the starting exposure times ranging from 0 to 2.5 ms. (c) is mirror flip sequences at the selected starting exposure time.

projector model for nonintrusive projection. To understand the overall characteristics of the color wheels without full access to a DLP chip and its controller, we applied the camera-based classification method proposed in [36]. This section briefly explains the color-wheel analysis steps and shows the results regarding our DLP projector model.

First, this analysis method requires a camera with an external trigger feature to synchronize it with a DLP projector. Synchronization between the projector and the camera is performed here by tapping the vertical sync signal (5 V, 60 Hz) from the computer to the projector and using it to trigger the camera. In addition, the aperture size of the camera must be increased whereas the shutter of the camera must be set to open for a very short period (i.e., 0.55 ms in our setup) in order to sense the fast characteristics of the color wheel properly with no effect from the ambient light. All these settings refer to the *initial camera setup* step written in Figure 10.

Second, we analyzed the overall sequences of the color wheel inside our DLP projector by projecting single-color images (corresponding to the colors of each available color wheel of the projector) at maximum intensity through all possible starting exposure times. Figure 11(a) was created by allowing the synchronized camera to sense these projected colors with different starting exposure times. Then, we analyzed detailed mirror flip sequences for all 256 values in

the selected color channel (i.e., red, green, or blue) within a narrow starting exposure period. From Figure 11(a), we selected the red channel, which is the first channel appearing in the sequences; other color channels can be selected too, but it will result in a longer synchronization delay time. Mirror flip sequences were then obtained by projecting uniform red images with intensity values ranging from 0 to 255, and with the starting exposure times ranging from 0 to 2.5 ms. Figure 11(b) was created by allowing the camera to sense these red projections. A starting exposure time of 0.8 ms was finally chosen because it provides the best distributed red ramps, as shown in Figure 11(c).

Up to this point, the *color wheel analysis* step written in Figure 10 is done and we are able to synchronize the camera with the projector at the appropriate starting exposure time. For our selected starting exposure time, the camera can only see the red light of the DLP projector (correspondences between projected red intensities and red intensities seen by the camera are shown in Figures 11(b) and 11(c)).

6.2. Color Classification and Color Conversion. In [36], the 256 red intensities in the selected time slot were classified into three sets: *WHITE*, *BLACK*, and *GRAY*. *WHITE* refers to colors whose projection fully turns mirrors inside the DLP projector and transmits lights toward a surface (i.e., bright red light as seen by the camera). *BLACK* refers to colors whose projection does not flip the mirrors and transmits no light toward a surface (i.e., no light seen by the camera). *GRAY* refers to unreliable states between *WHITE* and *BLACK*. By converting each projected color to its most similar color in the *WHITE* or *BLACK* set, the work of [36] can imperceptibly embed the binary pattern to the normal projection. Based on the same idea of three sets of colors, our nonintrusive projection is performed by projecting only the contents whose colors share the same classified set so that they blend across the projection area seamlessly as seen by the camera. In this way, the camera is deceived and cannot see the projected contents because they cannot be differentiated from their background.

Nevertheless, the method of projecting only colors whose red intensity contained in the specific set works only for a DLP projector model that does not have interdependent color channels in mirror flipping. Instead of depending on the red channel and risking effects from the other color channels, we propose a DLP-model-independent classification that involves all color channels (i.e., red, green and blue) of projected images. Each color in the RGB color space was projected onto the white surface (locating in a static environment) and sensed by the camera. The projected colors were then classified into the three sets previously discussed. These projection and classification refer to the last step of the offline processes according to Figure 10.

To use the three classified sets of colors during online nonintrusive projection, we applied an approximate nearest-neighbor search algorithm called best bin first (BBF) [37] so that an arbitrary color is converted into our preferred set and visibility of the projected contents (as seen by the camera) can be controlled. However, as mentioned in Section 6.1, the exposure time of the camera is significantly decreased

in this approach. Therefore, the camera cannot sense the environment properly unless there is light emitting from the projector. This effect can be seen in Figure 9(E) where no ambient light is seen by the camera and the environment is only revealed inside the projector frustum area. For, further environment analysis purpose, this means that we need to illuminate the environment while projecting nonintrusive contents. Hence, all projected colors must be converted to the *WHITE* set to ensure both seamless visibility in the camera image and proper environment illumination. The final result of our nonintrusive projection is shown in Figures 9(E) and 9(F). It can be seen that the Canny edge detection result is similar to that of a conventional vision-based system where there is no projection (as shown in Figures 9(A) and 9(B)).

7. Proof of Concept

To prove the proposed framework, we implemented a basic program in a small-scale laboratory setting to demonstrate the concept of real-time object augmentation in a 3-D environment using the procedural flow as illustrated in Figure 1. The projector and camera were colocated and then calibrated by projecting a known pattern onto a planar white surface placed at a distance of 60 cm from the front edge of the wooden base. The calibration was performed offline by manually locating 25 pairs of correspondences ($\delta = 25$ in (2)) and computed the homography matrix as described in Section 4.2.

In the program, 2-D Gabor filter [38], known to be useful in segregating textural regions, is used as the base detector and any area whose Gabor filter response magnitude is low will be considered the target object for augmentation. In other words, the target objects for this program are textural regions or surfaces that have either nonuniform reflections or discontinuities. Based on this assumption, we detected areas of the target object inside the projector frustum and drew bounding rectangles around each detected area using a randomly assigned nonintrusive color. The detection is performed here in a single-frame analysis style and the raw detection data is utilized without further refinement. The purpose of this program is to show that our framework enables a visual detector to be used in the same way as nonprojector systems and can solve the fundamental problems of a vision- and geometry-based pro-cam system regardless of 3-D shape of the object or a multidepth plane. Nevertheless, despite the simple object detection, this program completely neglects other vision-based algorithms for eliminating noises, identifying an object's shape, tracking the target object, and so forth. Therefore, existing noises are not eliminated and the target object may not be perfectly extracted.

Figure 12 shows the program in action with the yellow drawn over the camera image representing the multiscale Gabor detection result. The snapshot images of the actual surface were also shot from an asynchronous separate camera in order to show how a human actually saw the projected augmentations. Figures 12(a) and 12(b) show results on nonplanar surfaces whereas Figures 12(c) and 12(d) display results on slanted and angled surfaces. Finally, Figures 12(e)

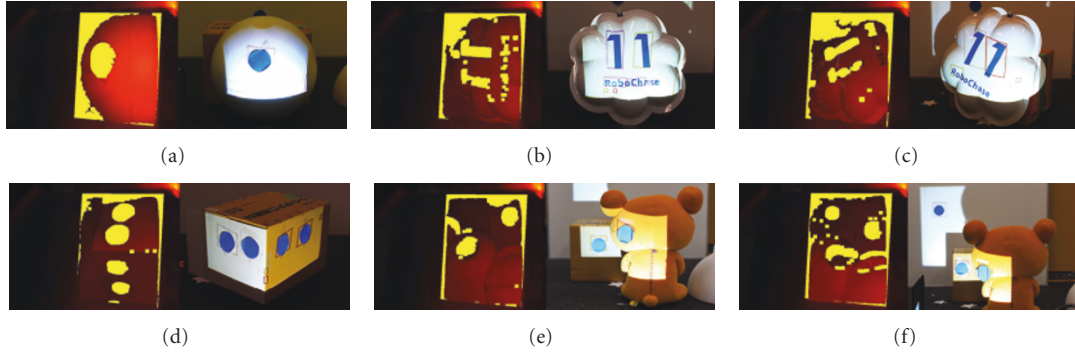


FIGURE 12: Demonstration of the proposed framework using 2-D Gabor filter as the base detector. Left images are images captured by the system camera with the multiscale detection result drawn in yellow. Right images are snapshots of the surface taken by a separate camera. All camera images were equally enhanced here for better visualization.

and 12(f) demonstrate results on multiple planes with different distances to the surface. Using the proposed framework, most blue-texture areas were detected and bounding rectangles were drawn around them regardless of their geometry and distance to the surface (the distance reflects the amount of light as seen by the camera as illustrated in Figure 5). Colors of the object also influence the detection result. As shown in Figure 12(e) the program failed to detect the brown stitch on the doll. This is because brown is the color that blends across the projector light (i.e., the red light illuminating the environment) when being seen by the camera (detailed explanation as written in Section 6). Therefore, the program could not distinguish it from the background in camera images.

The same procedure can be repeated when developing applications in a real-world scenario. Instead of the simple visual detector used in this paper, human detector, gesture recognition or motion analysis algorithms might be used to carefully refine the raw data and detect a user or interpret his actions accurately in real time. Final robustness of the application regarding object or human detection will depend on the selected visual detector and additional refinement algorithms. Our framework finishes its tasks here for solving problems of a pro-cam system, leaving problems of a conventional vision system to be solved by other appropriate vision-based algorithms.

8. Experimental Results

In this section, we discuss experiments conducted to evaluate the three proposed approaches. All experiments were performed using an HP Pavilion dv5 Notebook PC with an Intel Core2 Duo CPU P8600 running at 2.40 GHz. The following devices were used in the experiments: a TechSpec plate beam splitter 48904-J, an Acer P3250 DLP projector (Native XGA 1024 × 768 projection resolution with standard brightness at 2000 ANSI lumens), a Dragonfly Express camera (VGA 640 × 480 captured resolution), and an ELECOM VSP-A2 VGA splitter. The camera is equipped with a Tamron 13VM308AS lens, and the projector's focus was adjusted manually in all experiments.

Scene-Independent Pro-Cam Geometric Calibration. To determine the accuracy of the proposed pro-cam geometric calibration, we conducted experiments on both planar and nonplanar surfaces. The offline calibration was performed using only one sample image containing 25 calibrated points ($\delta = 25$ according to (2)). In the experiments, camera coordinates generated by our approach were compared with actual camera coordinates located manually. For the planar and slanted surface, five experiments were conducted with different distances of the projector, and each experiment was performed using 25 tested points (the number of tested points written here is not the δ value used in the offline calibration). The same experiment with 25 tested points was repeated on five nonplanar surfaces which are difficult to be calibrated in real time using other noncolocating designs. Figure 13(a) shows the experimental results on the planar and slanted surfaces whereas Figure 13(b) shows the experimental results on the nonplanar surfaces. Snapshots of the nonplanar experimental surface are captured by a separate camera and shown in Figure 13(c).

According to the experimental results shown in Figure 13(a), the proposed calibration provides the narrow range of geometric errors in both axes. In the fourth and fifth experiments, geometric errors are higher than those of the first three experiments. This is because the fourth and fifth experiments were conducted on a slanted surface and depth variation of the surface slightly affected accuracy of the calibration. Detail regarding this will be explained in the next paragraph.

According to Figure 13(b), the geometric errors for these nonplanar surfaces are similar to those of the planar and slanted surfaces, except for the third and fourth experiments. In the two mentioned experiments, the errors along the X axis are higher than the others, but still less than 3% compared with the width of the captured resolution. These increases of errors are caused by the significant depth variation of these two experimental surfaces; the variation makes the projected image become slightly distorted when being seen by the camera. Hence, points lying on the same straight line but different depth planes are not ideally collinear in the camera image and results in geometric errors

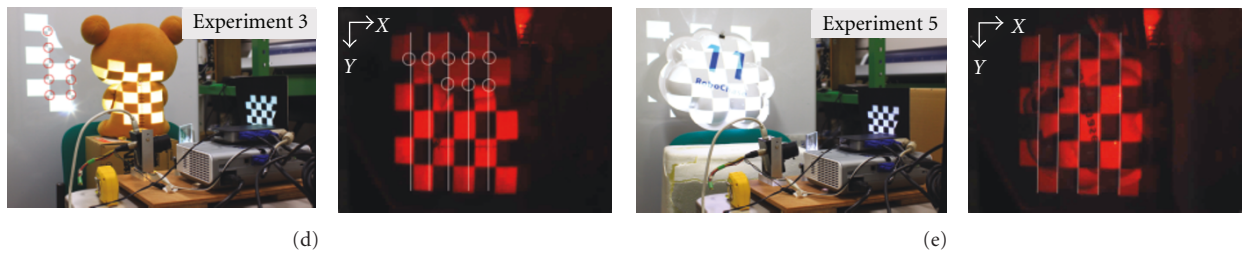
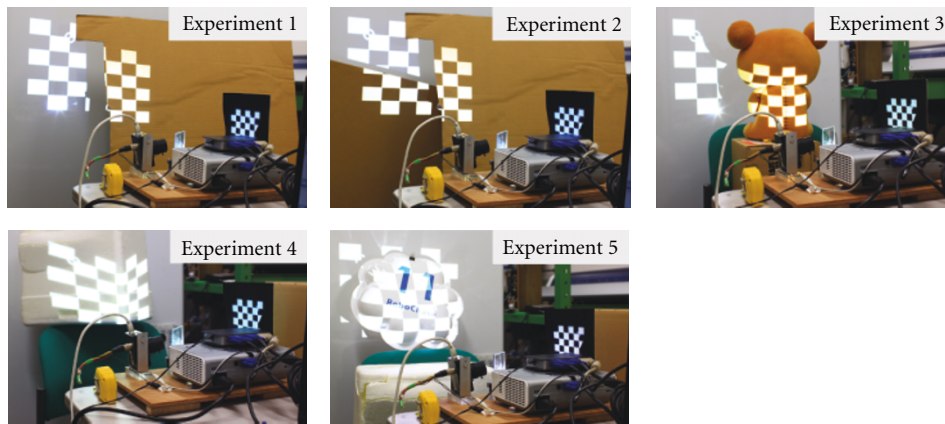
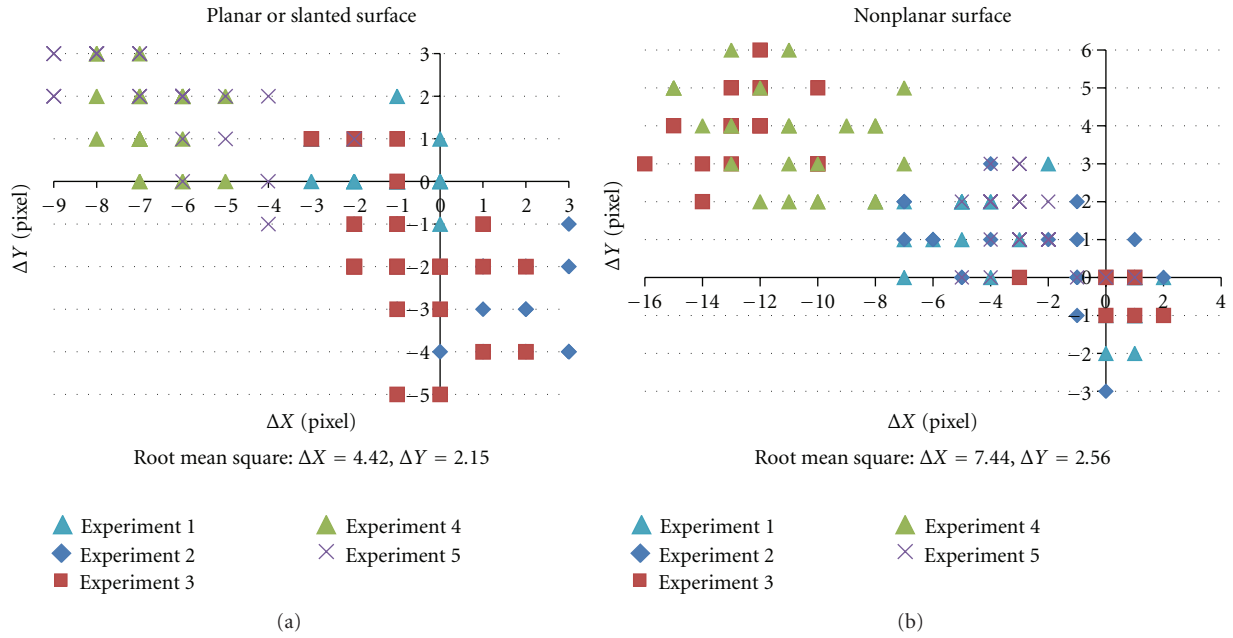


FIGURE 13: (a) Geometric error (camera coordinates) of the proposed pro-cam geometric calibration approach on planar and slanted surfaces. (b) Geometric error (camera coordinates) of the proposed pro-cam geometric calibration approach on nonplanar surfaces. (c) Snapshots of the nonplanar experimental surfaces. (d, e) are the captured images of the third and fifth nonplanar experiments, respectively.

of the calibration. This is shown in Figure 13(d) where there are two groups of tested points: 8 and 17 points with and without bounding circles, respectively. The first group of points lying on the whiteboard has small geometric errors whereas the second group of points projected onto the doll in front of the whiteboard provides larger geometric errors. However, if the depth variation of the projection surface is

not high, as shown in Figure 13(e), 3-D shapes of the surface barely cause distortion of the projected image as seen by the colocated camera.

In conclusion, the proposed calibration approach provides reasonably accurate results for the purpose of ubiquitous object augmentation, particularly when dealing with a surface whose geometries are not known. The accuracy of

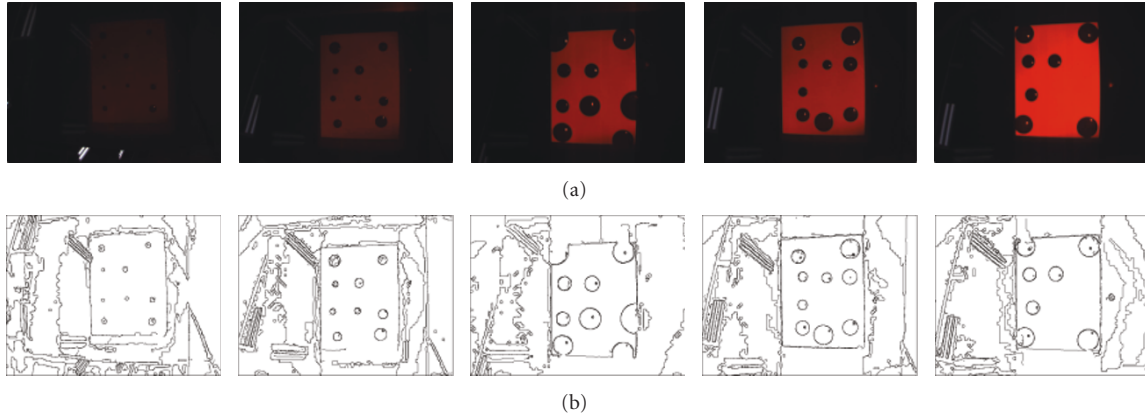


FIGURE 14: Images in the top row were captured by the proposed pro-cam design. Images in the bottom row result from Canny edge detection using the proposed multiscale visual detection scheme with parallel implementation.

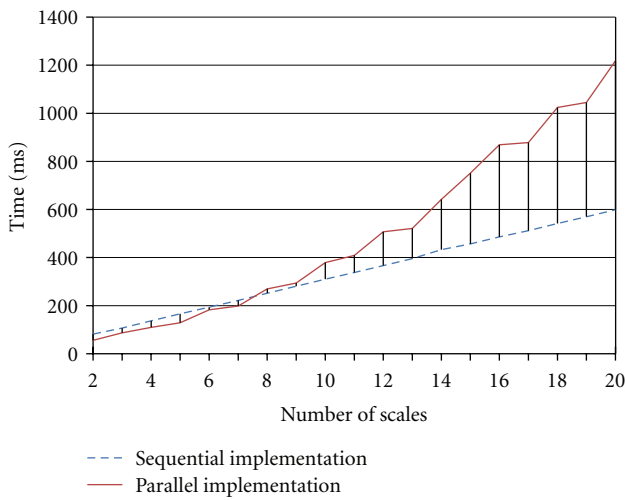


FIGURE 15: Relationship between the computational time (per one input image) and the number of scales (N) using the sequential and parallel implementation of the proposed detection scheme.

the calibration does not fall over time but may be slightly decreased when the projection surface has significant depth variation.

Multiscale Visual Object Detection. For the proposed detection scheme, first, we investigated its performance using the parallel implementation in differently illuminated images. Figure 14 shows the experimental results using Canny edge as a representative detector. All experiments share the same settings and no further parameter adjusting is performed in each experiment. According to Figure 14, it can be seen that the proposed scheme is capable of detecting true edges inside the projector frustum area in all images. Noise edges are visible inside the detected objects given the reflection of the projector’s light upon the object’s surface.

To determine the speed efficiency of the parallel implementation, Figure 15 shows the computational time used during the multiscale detection in the sequential and parallel

implementations versus the number of scales (N). Our parallel implementation used the OpenMP parallel programming architecture. Therefore, the parallel capability is limited to our Core2 Duo CPU. According to Figure 15, the times used by the sequential implementation increase linearly while the times seem to increase exponentially in the parallel implementation. The parallel implementation uses less time than the sequential does when the number of scales is less than 8. This can be explained by the concept of parallel programming in which the ratio of the number of data transfers to the number of executed commands must be small; otherwise, executing in a sequential manner on a single CPU is faster. From our experiments, the number of scales from four to six is adequate. Hence, using OpenMP programming with a recent processor providing more than two cores should allow for easy acceleration. Another alternative is to use GPU programming, but major reimplementations of the existing program may be required.

Nonintrusive Projection. To evaluate the nonintrusive projection, we randomly generated 64 colors and converted them into the *WHITE* set using the proposed color conversion approach. The converted image was then projected and sensed by the synchronized camera. Figure 16 shows the experimental results including conversion of the same image to the *BLACK* set for comparison. From images captured by the camera shown in Figure 16, our classification and conversion approach can perform efficiently and the surface remains well illuminated when projecting colors in the *WHITE* set. However, the projection area as seen by the system camera is not ideally seamless when projecting colors in the *WHITE* set. This is due to the range of the threshold set during the color classification (Section 6.2). By the way, these traces are not clear enough to intrude on visual environment analysis.

9. Conclusion

In this paper, we proposed a complete vision-based framework for real-time object augmentation applications using

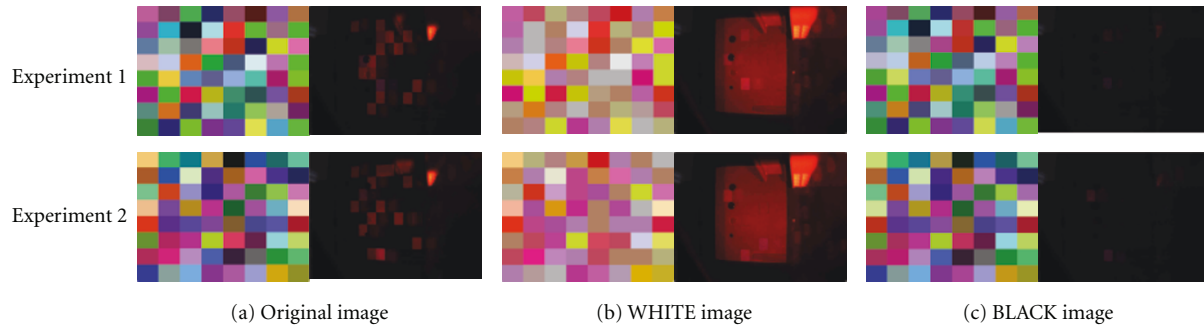


FIGURE 16: Color conversion using the proposed approach. (a) is the randomly generated 64 colors before conversion. (b, c) are (a) images after converted to the WHITE and BLACK set, respectively. Images in (b) represent the concept of nonintrusive projection used in this paper. Note that for each paired image, the right image shows the environment as seen by the synchronized camera when the left image is being projected from the projector.

a portable pro-cam device. Only the geometric aspect of the system is concerned in order to detect the target object and project real-time 2-D augmentation in the desired geometry. The framework consists of colocating the projector and camera using a beam splitter so that scene-independent geometric calibration can be accomplished using perspective transformation model. Projective interference problem is solved by synchronizing the camera with the DLP projector at a specific time slot, allowing real-time projection and real-time visual object detection to be done simultaneously without visually interfering with each other. Finally, the problem of variations of lighting introduced by our proposed design is solved using the multiscale visual detection scheme together with parallel programming.

The proposed framework is truly portable, requires no stationary device in the workspace, and attaches no marker to the target object or the projection surface. Additional algorithms can be added to allow more realistic and smooth object augmentation in an actual application.

References

- [1] P. Mistry and P. Maes, "SixthSense: a wearable gestural interface," in *Proceedings of the SIGGRAPH ASIA, Emerging Technologies and Sketches*, 2009.
- [2] R. Raskar, J. Van Baar, P. Beardsley, T. Willwacher, S. Rao, and C. Forlines, "iLamps: geometrically aware and self-configuring projectors," in *Proceedings of the ACM SIGGRAPH*, pp. 809–818, July 2003.
- [3] M. Lochtefeld, S. Gehring, J. Schoning, and A. Kruger, "ShelfTorchlight: augmenting a shelf using a camera projector unit," in *Proceedings of the 1st Workshop on Personal Projection (UbiProjection '10)*, 2010.
- [4] S. Borkowski, O. Riff, and J. Crowley, "Projecting rectified images in an augmented environment," in *Proceedings of the IEEE International Workshop on Projector-Camera Systems (PROCAMS '03)*, 2003.
- [5] J. C. Lee, S. E. Hudson, J. W. Summet, and P. H. Dietz, "Moveable interactive projected displays using projector based tracking," in *Proceedings of the Annual ACM Symposium on User Interface Software and Technology (UIST '05)*, pp. 63–72, 2005.
- [6] K. Hosoi, V. N. Dao, and M. Sugimoto, "CoGAME: manipulating by projection," in *Proceedings of the ACM SIGGRAPH, Emerging Technologies (SIGGRAPH '07)*, 2007.
- [7] M. Kanbara, A. Nagamatsu, and N. Yokoya, "Augmented reality guide system using mobile projectors in large indoor environment," in *Proceedings of the Workshop on Personal Projection (UbiProjection '10)*, 2010.
- [8] D. Cotting and M. Gross, "Interactive environment-aware display bubbles," in *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology (UIST '06)*, pp. 245–254, October 2006.
- [9] X. Cao and R. Balakrishnan, "Interacting with dynamically defined information spaces using a handheld projector and a pen," in *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology (UIST '06)*, pp. 225–234, October 2006.
- [10] D. Molyneux, H. Gellersen, G. Kortuem, and B. Schiele, "Cooperative augmentation of smart objects with projector-camera systems," in *Proceedings of the 9th International Conference on Ubiquitous Computing (UbiComp '07)*, vol. 4717, pp. 501–518, September 2007.
- [11] J. Schöning, M. Lochtefeld, M. Rohs, A. Krüger, and S. Kratz, "Map torchlight: a mobile augmented reality camera projector unit," in *Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems (CHI '09)*, pp. 3841–3845, April 2009.
- [12] M. Fiala, "Automatic projector calibration using self-identifying patterns," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '05)*, 2005.
- [13] R. Raskar, J. van Baar, and J. X. Chai, "A low-cost projector mosaic with fast registration," in *Proceedings of the IEEE Asian Conference on Computer Vision (ACCV '02)*, 2002.
- [14] R. Raskar and P. Beardsley, "A self-correcting projector," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. II504–II508, December 2001.
- [15] J. van Baar, T. Willwacher, S. Rao, and R. Raskar, "Seamless multi-projector display on curved screens," in *Proceedings of the Workshop on Virtual Environments (EGVE '03)*, 2003.
- [16] W. Sun, X. Yang, S. Xiao, and W. Hu, "Robust checkerboard recognition for efficient nonplanar geometry registration in Projector-Camera systems," in *Proceedings of the 5th ACM/IEEE International Workshop on Projector Camera Systems (PROCAMS '08)*, August 2008.

- [17] Y. Ding, J. Xiao, K. H. Tan, and J. Yu, "Catadioptric projectors," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshop (CVPR '09)*, pp. 2528–2535, June 2009.
- [18] A. Kushal, J. Van Baar, R. Raskar, and P. Beardsley, "A handheld projector supported by computer vision," in *Proceedings of the 7th Asian Conference on Computer Vision (ACCV '06)*, vol. 3852, pp. 183–192, 2006.
- [19] T. Johnson and H. Fuchs, "Real-time projector tracking on complex geometry using ordinary imagery," in *Proceedings of the International Workshop on Projector-Camera Systems (PROCAMS '07)*, 2007.
- [20] R. Raskar, G. Welch, M. Cutts, A. Lake, L. Stesin, and H. Fuchs, "Office of the future: a unified approach to image-based modeling and spatially immersive displays," in *Proceedings of the Annual Conference on Computer Graphics (SIGGRAPH '98)*, pp. 179–188, July 1998.
- [21] A. Grundhöfer, M. Seeger, F. Häntsch, and O. Bimber, "Dynamic adaptation of projected imperceptible codes," in *Proceedings of the 6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '07)*, November 2007.
- [22] H. Park, M.-H. Lee, B.-K. Seo, Y. Jin, and J.-I. Park, "Content adaptive embedding of complementary patterns for nonintrusive direct-projected augmented reality," in *Proceedings of the International Conference on Human-Computer Interaction (HCI '07)*, vol. 4563, pp. 132–141, 2007.
- [23] M. Waschbüsch, S. Würmlin, D. Cotting, F. Sadlo, and M. Gross, "Scalable 3D video of dynamic scenes," *Visual Computer*, vol. 21, no. 8–10, pp. 629–638, 2005.
- [24] S. G. Narasimhan, S. J. Koppal, and S. Yamazaki, "Temporal dithering of illumination for fast active vision," in *Proceedings of the 10th European Conference on Computer Vision (ECCV '08)*, vol. 5305, pp. 830–844, 2008.
- [25] S. J. Koppal and S. G. Narasimhan, "Illustrating motion through DLP photography," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '09)*, pp. 9–16, 2009.
- [26] Z. Wang, Z. Miao, and C. Zhang, "Extraction of high-resolution face image from low-resolution and variant illumination video sequences," in *Proceedings of the 1st International Congress on Image and Signal Processing (CISP '08)*, pp. 97–101, May 2008.
- [27] J. Lou, H. Yang, W. Hu, and T. Tan, "An illumination invariant change detection algorithm," in *Proceedings of the IEEE Asian Conference on Computer Vision (ACCV '02)*, pp. 13–18, 2002.
- [28] T. Tanaka, A. Shimada, D. Arita, and R.-I. Taniguchi, "Object detection under varying illumination based on adaptive background modeling considering spatial locality," in *Proceedings of the 3rd Pacific Rim Symposium on Image and Video Technology (PSIVT '09)*, vol. 5414, pp. 645–656, January 2009.
- [29] W. Maier, F. Bao, E. Mair, E. Steinbach, and D. Burschka, "Illumination-invariant image-based novelty detection in a cognitive mobile robot's environment," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '10)*, pp. 5029–5034, 2010.
- [30] P. M. Roth, S. Sternig, H. Grabner, and H. Bischof, "Classifier grids for robust adaptive object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '09)*, pp. 2727–2734, 2009.
- [31] K. Fujii, M. D. Grossberg, and S. K. Nayar, "A projector-camera system with real-time photometric adaptation for dynamic environments," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 814–821, June 2005.
- [32] R. Sukthankar, R. G. Stockton, and M. D. Mullin, "Smarter presentations: exploiting homography in camera-projector systems," in *Proceedings of the 8th International Conference on Computer Vision (ICCV '01)*, pp. 247–253, July 2001.
- [33] Open Source Computer Vision (OpenCV), <http://opencv.willowgarage.com/wiki/>.
- [34] Q. Yang, K. H. Tan, and N. Ahuja, "Real-time O(1) bilateral filtering," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR '09)*, pp. 557–564, June 2009.
- [35] B. Schwerdtfeger, D. Pustka, A. Hofhauser, and G. Klinker, "Using laser projectors for augmented reality," in *Proceedings of the ACM Symposium on Virtual Reality Software and Technology (VRST '08)*, pp. 134–137, October 2008.
- [36] D. Cotting, M. Naef, M. Gross, and H. Fuchs, "Embedding imperceptible patterns into projected images for simultaneous acquisition and display," in *Proceedings of the 3rd IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '04)*, pp. 100–109, 2004.
- [37] J. S. Beis and D. G. Lowe, "Shape indexing using approximate nearest-neighbour search in high-dimensional spaces," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '97)*, pp. 1000–1006, June 1997.
- [38] S. E. Grigorescu, N. Petkov, and P. Kruizinga, "Comparison of texture features based on Gabor filters," *IEEE Transactions on Image Processing*, vol. 11, no. 10, pp. 1160–1167, 2002.