

Embedding Imperceptible Patterns into Projected Images for Simultaneous Acquisition and Display

Daniel Cotting, Martin Naef, Markus Gross
ETH Zurich, Switzerland
{dcotting, naef, grossm}@inf.ethz.ch

Henry Fuchs¹
University of North Carolina at Chapel Hill
fuchs@cs.unc.edu

Abstract

We introduce a method to imperceptibly embed arbitrary binary patterns into ordinary color images displayed by unmodified off-the-shelf Digital Light Processing (DLP) projectors. The encoded images are visible only to cameras synchronized with the projectors and exposed for a short interval, while the original images appear only minimally degraded to the human eye. To achieve this goal, we analyze and exploit the micro-mirror modulation pattern used by the projection technology to generate intensity levels for each pixel and color channel. Our real-time embedding process maps the user's original color image values to the nearest values whose camera-perceived intensities are the ones desired by the binary image to be embedded. The color differences caused by this mapping process are compensated by error-diffusion dithering. The non-intrusive nature of our novel approach allows simultaneous (immersive) display and acquisition under controlled lighting conditions, as defined on a pixel level by the binary patterns. We therefore introduce structured light techniques into human-inhabited mixed and augmented reality environments, where they previously often were too intrusive.

1. Introduction

In a variety of augmented reality systems we have often wished to operate cameras and projectors simultaneously. Unfortunately, conflicting lighting requirements have made such systems very difficult to realize: cameras need brightly lighted environments, whereas projectors need dark rooms [1, 2]. An additional difficulty for cameras, especially for those performing 3D acquisition, has been the lack of strong features in many parts of the environment. Skin, clothes and furniture often image with nearly uniform surface structure, making depth acquisition that relies on stereo correspondence particularly difficult to per-

form. Using structured light to illuminate the scene solves this problem, however it is highly distracting and therefore not suitable for human-populated environments.

While wondering one day, "if only structured light weren't so distracting," one of our colleagues, Gary Bishop, pointed out that we live daily in a structured light environment, working under fluorescent lighting - it is just modulated sufficient fast that we are mostly unaware of it. That observation lead us to consider that micro-mirror based DLP data projectors also modulate light sufficiently fast that we are unaware of it. In addition, with them we can achieve precise light control of time and space in a way that is not distracting to users.

The initial imperceptible structured light (ISL) concept was demonstrated with hardware-modified DLP projectors in the Office of the Future [2]. As opposed to this earlier approach, our research is inspired by Henry Fuchs' suggestion to intelligently modify the intensity levels of the projected color primaries which allows for the use of conventional DLP projectors. We introduce in this paper an embedding method based on this vision of ultimate light control and embed structured light imperceptibly with off-the-shelf projectors, while also simultaneously projecting real-time color imagery.

2. Related work

Relatively little work has been done to integrate multiple cameras and projectors into a single system. blue-c [1] provides an immersive environment for virtual reality and collaboration, combining projection and acquisition. The National Tele-Immersion Initiative [3] employs 3D avatars, where the geometry is acquired by trinocular stereo [4]. A combination of depth acquisition and immersive projection was first proposed in the Office of the Future project [2]. Here, a proof of concept for the embedding of structured light into DLP projections was achieved by significant changes of the projection hardware, including removal of the color wheel and reprogramming of the controller. The

¹ Guest Professor, ETH Zurich, 2003-2004.

resulting images were greyscale only, and the implementation of this setting was impossible without mastering and full access to the projection hardware.

The embedding method presented in this paper allows simultaneous display and acquisition under controlled lighting conditions. Thanks to this new enabling technology a wide variety of applications in environments containing multiple projectors and cameras can be realized. Multiprojector environments have been frequently and extensively used in VR, AR, and tele-presence. A classical example is the CAVETM [5], a room sized projection cube of up to six walls. Since then, numerous commercial and non-commercial projection systems have been introduced, including Elumens VisionDomeTM or the multiuser responsive workbench [6]. The problem of projector calibration to non-planar surfaces has been tackled for instance by Raskar [7]. Likewise, smooth blending of multiple units requires thorough color adjustment [8].

We see our novel pattern embedding approach as an enabling technology for a wide range of computer graphics and vision applications. [9] and [10] provide a detailed discussion of general vision algorithms. Presenting the entire field of applicable structured light algorithms is beyond the scope of this paper. As a proof of concept, we have implemented a Gray code depth reconstruction [11] as well as a single-shot method by Vuylsteke and Oosterlinck [12].

3. Embedding imperceptible patterns

The underlying technology of our new embedding technique is the micro-mirror based data projection technology, called Digital Light Processing (DLP). Therefore, we will devote the next section to the basics of the DLP technology. Since our approach exploits the pulse width modulation method that DLP projectors employ for displaying gradations of intensity values, we will present an analysis of the projection structure, and elaborate on the embedding possibilities we get by only controlling the input data of the projector without modifying the projection hardware.

3.1. DLP mode of operation

The heart of a DLP projector is a large CMOS IC, called DMD, the top surface of which is composed of a dense array of tiny $16\mu\text{m} \times 16\mu\text{m}$ mirrors, each corresponding to a single pixel in the 2D image to be projected. Each of these mirrors can be tilted electrostatically to one of two stable positions by control circuitry on the chip. A projected image is achieved by reflecting a light source off this array of micro-mirrors through a lens onto the projection surface. A pixel's projected intensity is determined by the percentage of time that its mirror is tilted toward the screen

rather than away from it. Color images are achieved, in all but the most expensive projectors, by placing a rapidly rotating color wheel with RGB sectors in the optical path and synchronizing the mirror-flips of each primary color with the period during which the corresponding filter is in the optical path.

3.2. Time-slot pattern embedding

The DLP micro-mirrors switch between their two possible orientations very quickly, within approximately $10\mu\text{s}$. When the pixels are set to dark, DLP projectors transport hardly any light onto the screens, hence the 1000 to 1 contrast ratio specifications of many DLP projectors. Since in its reflecting orientation, a mirror is throwing full brightness, we can use even a short time (e.g. $125\mu\text{s}$) to sufficiently expose a video camera sensor.

This high contrast ratio achievable at short exposure times allows our simultaneous acquisition to take place in a very short time-slot lying below the temporal resolution of the human eye. If we gain direct control of the micro-mirrors, we can light the scene during the exposure time with whatever pattern of light is most advantageous for a chosen acquisition system, without the user noticing the pattern in his application-specific projection.

To achieve this goal, the new scheme introduced in this paper does not rely on modifying the projector [2], but rather initially measures the existing mirror-flip sequences for different color values and then for each frame chooses pixel values that have the sequences that we want.

Concerning the projector interface, a digital video interface (DVI) is recommended over analog VGA, because we want to ensure that the precise pixel values we carefully select do not get changed by the time the projector's internal processor chooses the pixels' mirror-flip sequences. With analog video, a voltage-encoded value is transmitted to the projector and redigitized inside it; a process which does not guarantee that the projector receives the precise values originally sent by the graphics board. Nevertheless, we have also successfully used DLP projectors having analog input only for our method.

3.3. Alternative approaches

Texas Instruments has made available DMD discovery kits (<http://dmddiscovery.com>) to facilitate experiments with DLP micro-mirrors. While these kits give access to the DLP chip and its controller, they do not provide all the components necessary for building a projector. Additional engineering is required, whereas our method can be replicated easily by anyone, without hardware development, and should also work in the future with new, improved projectors and new cameras, obviating redesign effort. Fur-

thermore the costs for the discovery kits are much higher than the price for an off-the-shelf projector, rendering our chosen approach even more attractive.

Another alternative allowing invisible embedding of patterns is to use infrared (IR) projectors for generating the imperceptible structured light images. This approach is usually restricted to static patterns, and unfortunately the only IR projectors, to our knowledge, are made to provide test images for IR sensors for highly specialized markets. In contrast, DLP projectors are widely available. In addition, the usage of our embedding approach makes even more sense in settings where simultaneous visible projections are desired, because then only a single projector is needed for all the projection tasks, thus minimizing hardware costs.

3.4. Measuring a projector's mirror sequences

Since our embedding approach requires knowledge of the mirror-flip sequences, we need a setup to measure the light pulses for an unknown projector model. In a first approach, we use an experimental setup as shown in Figure 1: We place a photo transistor on the projector's display surface and measure its output with the help of a digital oscilloscope, which sends the collected data to a PC. Connecting a common synchronization signal to the oscilloscope and the projection allows precise triggering to measure from the beginning of each frame. Note that all the measurements can also be done with a simple camera setup as described in Section 3.6

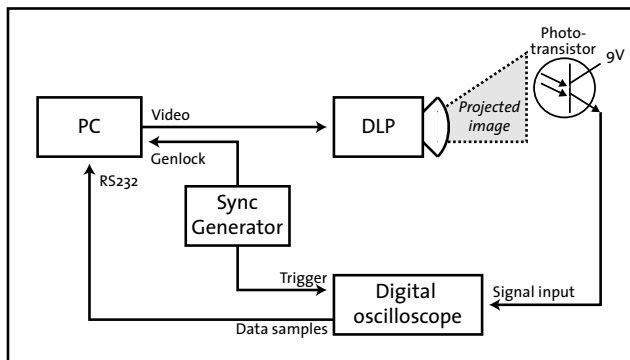


Figure 1: Initial setup to measure a projector's mirror sequences.

Figure 2 shows the mirror-flip sequence for one particular RGB value, (223,47,128). Notice that for the intensity level 223 the mirror is *on* most of the time, for level 47 *off* most of the time, and for intensity 128 *on* about 35% of the time (this projector's output response is not linear due to gamma correction). Notice there is a fourth sector in the color wheel of the measured projector, called clear, designed to boost light output at the expense of color

purity. This sector is not used in our example. Notice also that there are two RGBClear cycles in each 60 Hz frame, presumably to reduce perception of color flicker. The darker gray values in Figure 2 are due to the mirrors not being consistently *on* or *off* for certain values.

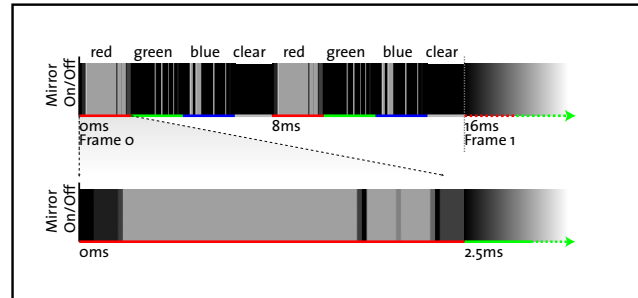


Figure 2: Mirror flip sequence for RGB value (223, 47, 128) during one frame. The clear segment is not used.

Several potential problems have to be addressed to reliably embed patterns into the mirror-flip sequences. We need to know whether the mirror-flip sequences vary over time and space. Furthermore, we need to explore if the sequences in one color sector of a pixel are independent of the values of the other primary colors of this pixel. Although much has been written about DLP designs over the years, specifics about any particular current projector are difficult to obtain, and no details about the firmware implementations are revealed. We therefore measured multiple sequences of a representative set of pixel values at different places on the display surface, for multiple frames, and while changing the values of other primaries of the same pixel. Our current results with several projectors synchronizing to the input signals (Infocus LP120/LP330/LP335 and projectiondesign F1) are the following:

- A certain amount of dithering occurs, but not enough to overwhelm the general mirror characteristics of most pixel values.
- Display during the clear sector has a serious deleterious effect: It introduces an interdependence between RGB components of the same pixel. For some pixel values, even a small change in the value of one primary causes a change in the mirror sequence of the other primaries - presumably to compensate for the change in the amount of light of that primary projected during the clear sector.

The complications due to the clear sector display made us use the projectiondesign F1 projector, which allows menu selection between high brightness and photographic modes. In photographic mode, the projector does not produce any light during the clear sector (of course at the expense of lower overall brightness), and no significant interdependence of the primaries can be found. The following experiments reported in this paper are therefore all conducted with the F1 projector.

3.5. Classification of mirror sequences

Figure 3 lists the mirror sequences for all 256 values of red for the first red phase in each frame. Observe the period from 0.4 ms to 0.5 ms at the start of the red sector: virtually no mirror flips occur during this period; the mirrors are either *off* during the entire period or *on* during the entire period. Thus, we can group all 256 red values into three classes: *black*, *white*, or *unreliable* (those that are dithered or otherwise not reliably *black* or *white*). Let us call the chosen period from 0.4 ms to 0.5 ms *binary image exposure period* (BIEP). All the following considerations will take place in this precise time-slot of the red channel. Note that other periods with similar properties exist, but in practice they are too short for reliable camera exposure.

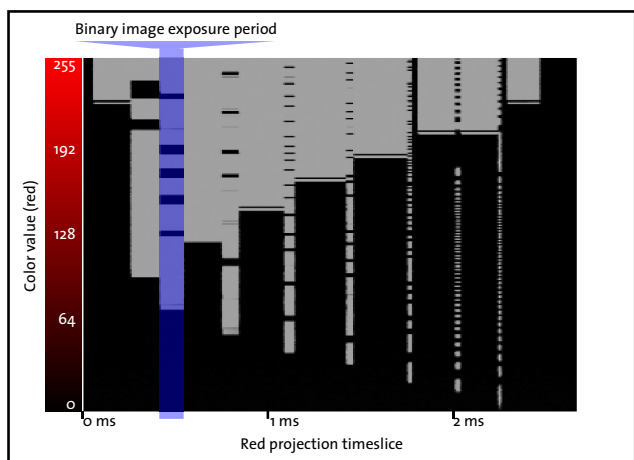


Figure 3: Measured mirror flip sequences for all 256 values of red (for projectiondesign F1 projector). Darker gray values indicate times when mirrors are not consistently *on* or *off*.

Since the three color channels are independent from each other for the chosen projector, the same measurements can be done for the red, the green and the blue channel separately and without considering the other primaries.

3.6. Camera-based classification

Although the measurement procedure with the photo-transistor and the oscilloscope gives us detailed insight into the structure of the mirror-flip sequences, it is not well suited for the color classification, because it requires hardware not everyone may have access to. We have therefore devised an approach to create the classification with a simple camera, the component that is usually already available in a system for simultaneous projection and acquisition (see Figure 4).

By studying Figure 3 we realize that we can acquire all required information by projecting a red-ramp with intensity values from 0 to 255 through all possible starting expo-

sure times, where each image will result in a column of Figure 3. In fact we can do this classification faster with a two-stage process. First we scan through all starting times, measuring the lowest pixel value with a near-white image. We thus find the optimal interval during which the ramp of pixel intensity values is perceived by the camera as having the greatest range of near-white intensities. We then choose this exposure time and measure the image values for all possible pixel intensity values of the corresponding ramp.

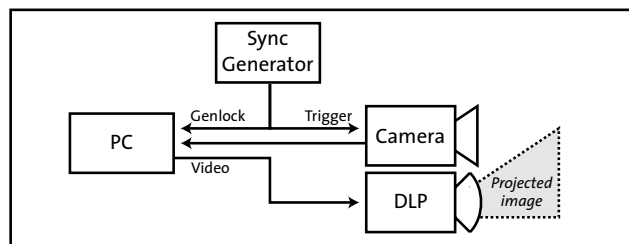


Figure 4: Setup for the camera-based mirror-flip sequence classification.

3.7. Sampling interdependent color channels

Projector models having interdependent color channels require an extended version of the camera-based classification method mentioned above. We have successfully sampled a subspace of the entire color space of one of these projectors and used the classification of this subspace for imperceptible embedding of our patterns. The color classification procedure is fully automatic and only takes a couple of minutes for each new and unknown projector.

3.8. Embedding algorithm

Consider exposing a synchronized video camera during the binary image exposure period (BIEP) introduced in Section 3.5. Since the acquisition camera is not exposed outside of this period, it is not affected by the actual content of the image projected, but only by what is projected during the BIEP. If we wanted the camera to see no light at all, we could alter the image sent to the projector, in software, to use only those values of red contained in the *black* set. Alternately, if we wanted to illuminate the scene brightly, we could modify the image so it contains only red values in the *white* set. Better yet, since we can adjust each pixel independently, if we want the camera to be exposed while the scene is illuminated by some structured light, such as an array of bars, a checkerboard, or a set of crosshairs, we can adjust each pixel in the image to be projected so that it comes from the *black* or *white* set, whichever is specified by the corresponding pixel in the binary control image. Figure 5 shows these sets and the adjustments to a pixel value to move it to the nearest *black* or *white* set.

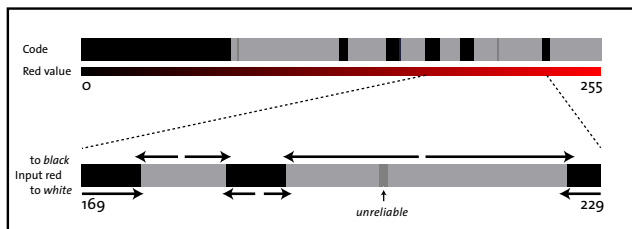


Figure 5: Red mirror positions during the binary image exposure period (BIEP): *white*, *black*, or *unreliable* (dark grey). Arrows indicate the direction of the moves to adjust an input pixel value to the intensities with the desired BIEP values.

One drawback of this scheme is that each red pixel luminosity can now come from only approximately half the original set, so projected images will show reduced tonal resolution. However, we are able to improve perceived colors by employing a slightly modified version of error-diffusion dithering, as described in many textbooks [13]. At each pixel, the decision is first made whether or not the pixel needs adjustment at all. If it needs to be adjusted, we first change it to the nearest pixel value with the desired binary value in the BIEP (see arrows in Figure 5). We then take the resulting difference (the error), and propagate it to two neighboring pixels (to the right and below) that are not yet processed. Figure 6 depicts a source image and the result after embedding a specific pattern.

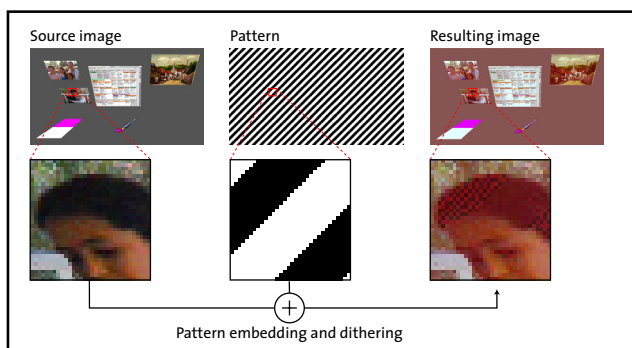


Figure 6: Embedding a binary image into a user-defined full-color image. Thanks to the ability of human perception to quickly recover color constancy, the color shift which is noticeable in the figure does not appear as pronounced in actual projections.

The result reveals a tonal shift due to the limited number of available red levels with low intensity values. We found this effect to be significantly less perceptible in the final projected images (see Figure 7) as opposed to images in print or on a computer display.

More faithful color reproduction at the expense of a reduced dynamic range can be achieved by compressing all color channels to the maximum range of the color channel that carries the embedded pattern (see Figure 8). With the tested projectors, this approach results in a dynamic range reduction of about 40 to 50 percent.

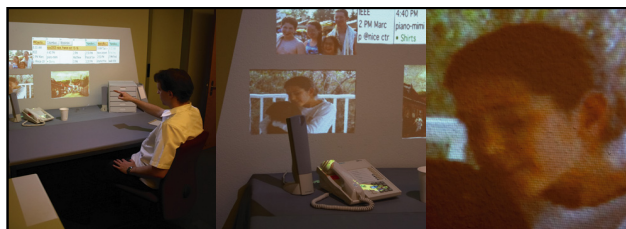


Figure 7: Three different levels of magnification of the projection in operation. Note that the dithering is completely invisible in actual scale, and no visible hint of the embedded binary image can be found. In this example the embedded pattern is used for depth reconstruction.

Color representation with an almost perfect range can be achieved by only embedding the pattern into areas of the full-color image that show a certain minimum color intensity, thus avoiding the mapping of dark values to lighter ones which was mainly responsible for the color-shift or the reduced dynamic range. This selective embedding approach works with pattern extraction algorithms having the capability to detect areas not containing any valid pattern. In order for every pixel to contain its corresponding pattern value at a certain point in time, the projected color imagery needs to be dynamic with lighter parts eventually covering the entire scene.

In case future DLP chips in projectors allow precise mirror control at the application level, optimally designed flip sequences can easily enhance the visual quality of our approach.

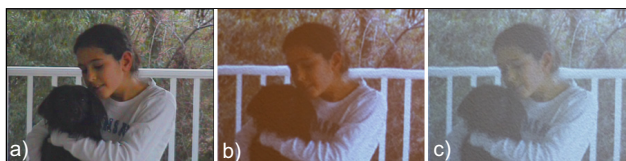


Figure 8: Red shift vs. reduced dynamic range. a) presents the raw image to be projected. b) results from only compressing the red channel during projection. c) shows an image with all channels compressed equally during projection.

4. Hardware system

Based on the ideas presented in Section 3, we have implemented a prototype system that concurrently displays a user application and provides acquisition using off-the-shelf components in a standard office environment.

4.1. Module configuration

Our prototype system operates with two modules, each consisting of a PC (Dell Dimension 8300) with an nVidia Quadro FX3000G graphics board, a projector and a grey-scale camera. The projectors and cameras are mounted to the ceiling using an aluminium rig as shown in Figure 10.

We use the genlock capability of the graphics boards to work with overlapping projection regions and to work with external synchronization options. For a setup without overlapping projection regions, a less expensive graphics board without the genlock capability is fully sufficient.

The two modules cover the working area of a user's desk (approximately 1.5m x 1.5m): Module 1 covers the volume from the wall above the desk down to the back half of the desk. Module 2 covers the user's body and the front part of the desk.

4.2. Projectors

In our setup we use two projectiondesign F1 projectors with SXGA resolution (1280 x 1024). They are provided either with a zoom lens or a wide angle fixed focal-length lens. We chose wide angle lenses so that the projection volume, with ceiling-mounted projection in our office, covers our desk and lights the user over a natural range of positions while sitting at the desk.

4.3. Cameras

We use Point Grey Dragonfly greyscale video cameras with an IEEE 1394 (Firewire) interface. These allow external synchronization, and they provide software control of various parameters through the IEEE 1394 interface. The most useful parameter for our purposes is the synchronization delay control, which defines a time delay between the external synchronization trigger and the actual start of the camera exposure period. This programmable delay allows us to exactly synchronize the camera to the BIEP without additional trigger delay hardware.

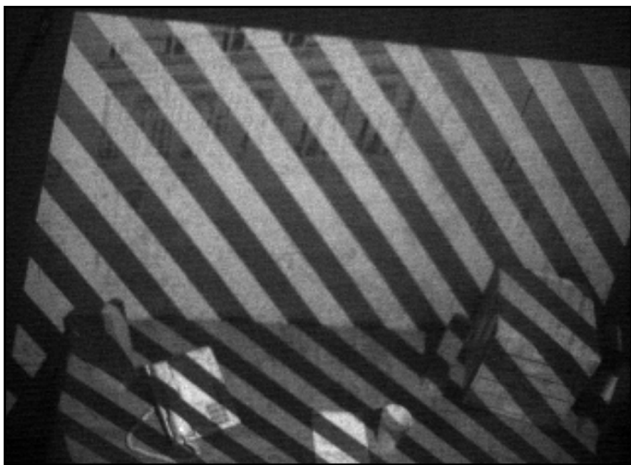


Figure 9: Image from a camera exposed during the binary image exposure period (BIEP).

We fine-tune this camera delay value by first projecting a still image with a binary pattern embedded in the desired

BIEP (see Section 3.5), and then scanning a broad range of camera delay values, capturing a camera image with each delay setting and calculating the contrast in that image. We select the synchronization delay value resulting in the highest-contrast camera image. This automatic synchronization procedure takes less than two minutes.

We get some cross-talk from the adjacent mirror flips due to the minimal camera exposure time of 125 μs, which is slightly larger than the duration of the BIEP. We also have to keep the lens wide open (F1.2) to get enough brightness at acceptable noise levels. Even though faster and more sensitive cameras would result in higher contrast and an increased depth of field, we found the resulting images to be fully adequate for our purposes. Figure 9 shows an image captured with an embedded stripe pattern.

4.4. Synchronization

We currently have the choice between two different options for the synchronization of cameras and projectors. Either we use an external sync generator and graphics boards featuring genlock (see Figure 4), or traditional (inexpensive) graphics cards and a device to tap the vertical sync signal of the projected imagery.

The first approach is straightforward and can be realized with off-the-shelf components only, using a commercial sync generator or a self-implemented microcontroller-based sync generator. This is basically the classical solution of installing a 60 Hz sync generator and distributing its signal to all cameras and all projectors.

The second approach is a little bit more involved since it requires a device for tapping the vertical sync signal. One can then synchronize a camera to its projector by connecting the projector's vertical sync signal to the external synchronization input of the camera. Using DVI instead of analog video in this setting is slightly more complex, because there is no isolated vertical sync signal going to the projector. We solve that problem by obtaining a traditional vertical sync signal by tapping the appropriate pin inside an off-the-shelf DVI repeater as shown in Figure 10.

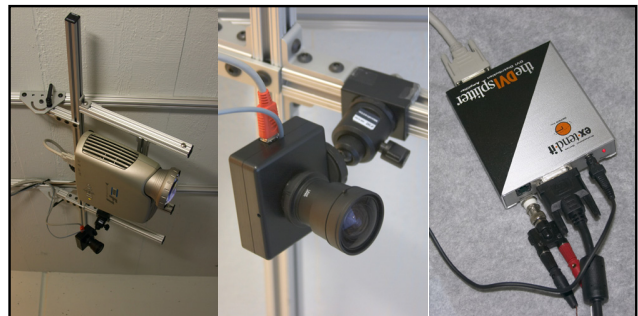


Figure 10: Hardware components: Projector, camera, and DVI repeater with tapped vertical sync signal.

4.5. Calibration

The cameras and projectors must be calibrated intrinsically and extrinsically with relation to each other, and in case of setups consisting of more than one module, with respect to a common global world coordinate system.

The calibration routine is embedded into the controller software, providing a user-friendly, mostly automatic procedure. Both intrinsic and extrinsic camera parameter estimations are based on Intel’s Open Computer Vision library (www.intel.com/research/mrl/research/opencv).

For projector calibration, a checkerboard pattern is projected onto two previously calibrated planes, resulting in two sets of related 3D points that enable us to calculate projector position, orientation and frustum with adequate precision. We currently ignore lens distortion errors of the projector.

5. Proof-of-concept software

The software system presented in this section is used as an illustration of various applications that can be successfully implemented with our imperceptible embedding of patterns, the main contribution of this paper. We do not aim at presenting new state-of-the-art structured light algorithms, but rather limit ourselves to comprehensive and illustrative examples, which are combined into a simple 3D paint application. This application consists of two modules:

- Module 1 constantly tracks the surface of the desk and the wall, sending the distance values to the renderer, which displays 2D images and 3D models onto the currently-measured surface. The renderer warps each image to account for the surface location and to ensure that the image appears correctly from the user’s current head position. Module 1 also executes the simple painting application described in Section 5.4.
- Module 2 constantly tracks the user’s head and hand positions and sends these values to module 1. At present, module 2 renders a simple flat field embedding visual feedback of the tracked hand position.

The software architecture for demonstrating our new embedding approach uses a modular setup to implement the individual components of the image generation and acquisition pipelines. Depending on the desired task of both acquisition and projection modules in our physical setup, the pipelines can be reconfigured on the fly. We have implemented components for depth acquisition, for projection undistortion, and for head- and hand-tracking including visual feedback. Figure 11 depicts two pipelines including the data flow between the components. On each module, the pipelines are controlled and synchronized by networked processes called *Flow Controllers*. The compo-

nents of the pipelines are implemented as Windows Direct-Show filters, thus enabling simple reuse for a plentitude of applications.

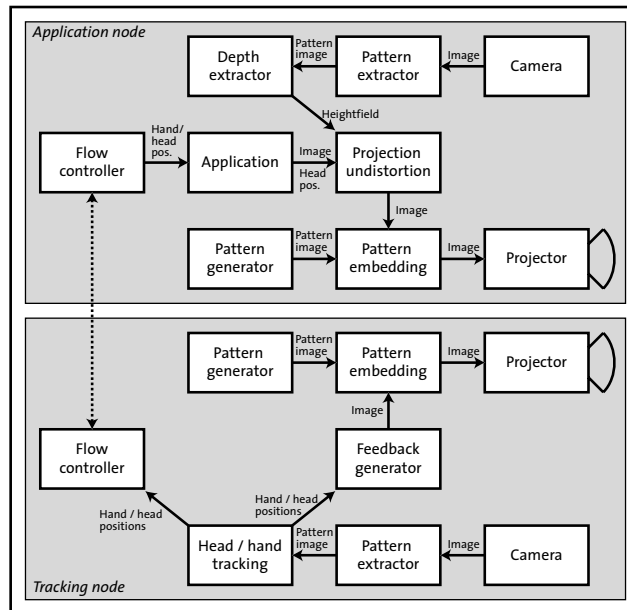


Figure 11: Software architecture. Currently target tracking is executed on one PC-projector-camera module, surface tracking on the other.

5.1. Depth acquisition

We have implemented two examples of depth acquisition algorithms to illustrate the suitability of our embedding for surface acquisition based on structured light: a basic Gray code surface extraction and a single-shot method based on previous work by Vuylsteke and Oosterlinck [12].

Let us briefly present the first, very simple, but illustrative approach, where the depth field acquisition is implemented using a simple Gray code based algorithm. The *pattern generator filter* sends a sequence of eight stripe pattern images, each representing a bit in the Gray code, to the *pattern embedding filter*. Additionally, two reference images (all white and all black) are embedded in order to simplify the pattern extraction. The *pattern extractor filter* decodes this bit pattern from the sequence of acquired camera images using a standard thresholding algorithm. The intersection between the ray through the camera pixel and the corresponding projector plane as defined by the extracted Gray code defines the position in 3D space [11]. Figure 12 shows a visualization of the three stages: a single Gray code image in the sequence, the decoded gray code values, and the computed depths.

The stability of the acquired depth field is increased by only reconstructing depth values at pixels where the same Gray code values have been detected over multiple

sequences. This both eliminates the effects of camera noise as well as transient changes of the depth values, e.g. caused by a user moving in front of the projector. In our initial prototype implementation, we have set the time to acquire a stable depth field to approximately ten seconds. Changes remaining static in the scene for a longer period of time are then updated at about the same rate, whereas transient changes are ignored as mentioned before. As a final step, a small Gaussian filter kernel is applied to the depth field to minimize the stair-step-effects caused by the finite resolution of the Gray code.

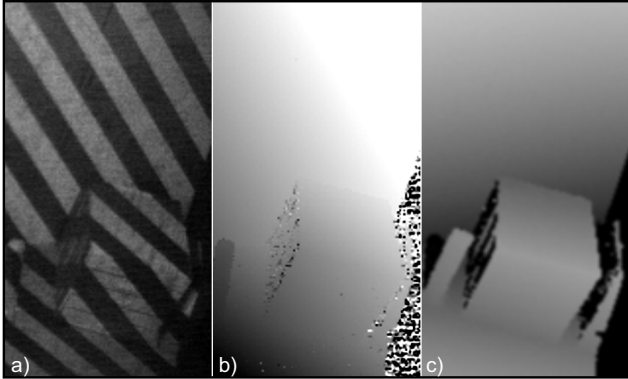


Figure 12: Depth acquisition: a) Image of one Gray code pattern in the sequence. b) Extracted Gray code value. c) Depth image.

A single-shot method can be used for applications wishing to acquire dynamic, moving scenes. Figure 13 shows a color-coded output of a depth map created with an initial implementation of the chosen sample algorithm [12].

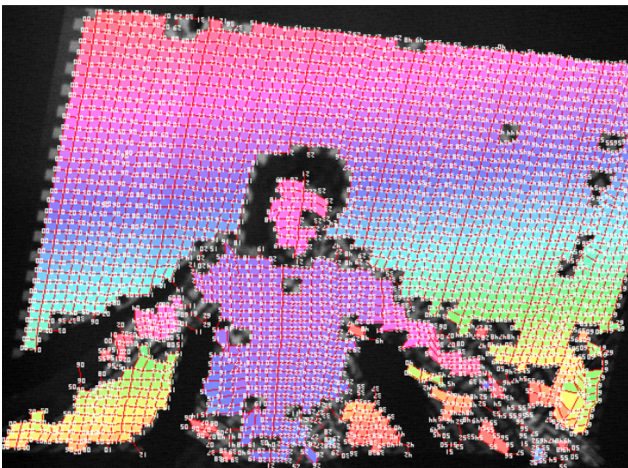


Figure 13: Color-coded depth of a moving person acquired by the chosen single-shot method.

Although we have only presented two well-known algorithms, we expect our embedding method to work with any of the contemporary state-of-the-art binary structured light methods.

5.2. Adaptive display onto non-planar surfaces

The depth extraction capability of our system enables us for example to deal with non-planar projection surfaces. This is a typical situation in an office environment with cluttered desks, bookshelves, drawers, etc. There are several standard methods for a projection display to deal with objects on the desk: either ignore them and therefore get distorted images, or be aware of them and don't project imagery onto them, or integrate the objects into the display scene as part of the projection surface, which is our approach.

In our prototype system, we use the depth extraction capabilities to frequently update the geometry of the projection surface. We then use a standard two-pass rendering method [2]: The image from the user's point of view is rendered into a texture. Then, the surface geometry is rendered as a regular grid from the projector's point of view, with the application texture applied using projective texturing. Holes in the reconstructed depth field are filled by interpolation.

5.3. Target tracking

Our sample *tracking filter* detects the user's hand and head position using a simple and efficient single-shot tracking method in real-time (at 20 fps, our camera frame rate).

The top projector is used as a lamp, illuminating the user and the working area. The light has an embedded horizontal stripe pattern projected onto the desk (tracking region) and a constant *white* code for the head region. The code is completely invisible to the user.

The initialization step consists of taking an averaged sample image showing the empty table, which is later used as a reference image. The hand tracking then works by detecting both the hand and its shadow in the tracking region. Image regions that are darker than the reference frame are classified as *shadow*, whereas brighter regions are classified as *hand*. The extremal positions (i.e. the positions closest to the wall) for both the hand and the shadow are extracted from these classified regions. The projector scanline projecting at the table location of the shadow extremal position can easily be reconstructed by counting the number of stripes until the detected extremal point is reached. The intersection of the corresponding 3D projector plane and the camera ray corresponding to the detected 2D hand point returns the location of the hand in 3D space. The regions and the detected extremal points are depicted in Figure 14.

The head position is defined as the brightest spot inside the head tracking region. The current prototype implementation assumes head motion on a constant plane in 3D space. It can easily be extended towards a real 3D tracking

by projecting a structured light pattern into the head tracking region, enabling additional depth reconstruction similar to Section 5.1.

Both hand and head positions are integrated over time to provide a smooth motion path to the application code. In our sample application, the user receives feedback on the detected hand position both by illuminating the finger using the top projector and by a cursor inside the application area. The detected head position is fed into the *projection undistortion filter* for a 3D image projection with correct perspective.

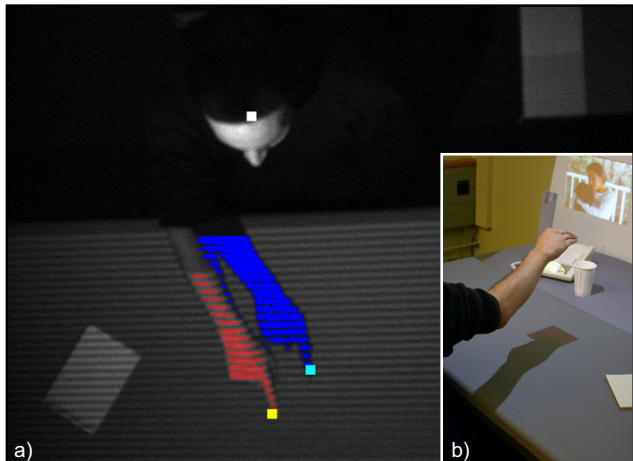


Figure 14: Tracking system at work: a) Color-coded camera image showing the hand shadow (blue), extracted image of the hand (red), finger tips in the shadow (turquoise) and on the hand (yellow). b) Visible image of the same scene. The user's head and hand can be tracked imperceptibly, allowing encumbrance-free hand-guided interaction and encumbrance-free head-tracking. Note the lack of any discernible artifacts of the embedded binary image.

5.4. Integrated application

We demonstrate the above tracking and display capabilities by integrating them into a simple 3D paint program. Figure 15 illustrates the application being used. It is implemented using the data flow presented in Figure 11. The user's hand and head are simultaneously tracked for use as the brush location in 3D space and for perspective correct rendering respectively.

The paintbrush releases circular “splats” at the current hand position. They are blended together for a smooth look. The size of the splats is defined by the speed of the brush, always resulting in a continuous, smooth line in space. The application state, including color selection, is controlled using simple voice commands for keyboard-free operation.

The display is customized to the user's head position so the user can observe the current 3D painting from a variety of viewing positions by direct head movement.

We have implemented additional sample applications that render movies or standard Windows applications onto arbitrary geometry in real-time.

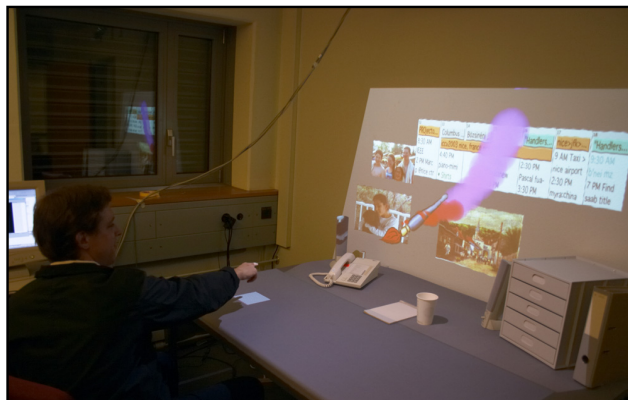


Figure 15: Paint program in action. The acquisition loop operates in parallel with display, monitoring changes in the 3D display environment and supplying the renderer with constantly updated estimates of the 3D location of every pixel of the display surface. The renderer can generate images properly warped to a desk, desk objects or walls with the correct geometry as seen from the user head's point of view.

6. Conclusion

In this paper we have presented a novel approach for imperceptible embedding of binary patterns into conventional unmodified DLP projectors, allowing simultaneous (immersive) display and acquisition under controlled lighting conditions. Our effective method will be an enabling technology for a wide range of computer graphics and computer vision algorithms. With its help, a wide range of binary structured light algorithms can be transferred to mixed and augmented reality systems, where visible pattern projections usually are too intrusive and severely limit the usefulness of the systems or the user experience. As a proof of concept we have implemented a set of sample algorithms clearly demonstrating the versatility of our imperceptible embedding. Application domains range from tracking over 3D depth extraction to perspective correct projection onto arbitrary changing geometry. While most of these applications certainly are possible without imperceptible structured light, our enabling technology makes them much easier and more reliable.

7. Future work

Scalability. Although our prototype implementation only consists of two modules, each with a PC, projector and camera, it appears straightforward to add more modules to achieve an increasingly immersive environment. The common issue is the one of light control: When an algorithm

wants a camera to take an (imperceptible) look at the scene, it wants to impose its own structured light pattern on the scene and be assured that no illumination from other projectors will interfere. This can be guaranteed by appropriate time-division and frequency-division multiplexing approaches. Furthermore, for the creation of large seamless displays, issues like geometry alignment, color balancing and gamut matching have to be addressed, taking into account the constraints resulting from our embedding procedure.

Extended structured light. In addition to currently available structured light acquisition techniques that can be implemented using our embedding method, we would like to use the possibilities offered by the simultaneous projection of different patterns using different time slots.

Adding stereo. A major capability currently lacking in our system is stereo projection. Of the many possibilities for achieving stereo, we hope to adopt a method of stereo that will also enhance the system's acquisition capabilities. For example, we have begun to consider a variation of the stereo approach used in [1]. They employ actively switched stereo glasses worn by the user and two (LCD) projectors for each wall. A shutter in front of each projector blocks a projector's light from the screen when the user is looking at the image from the other projector.

Continuous, fully-automatic calibration. We would like to eliminate our need for initial calibration. We are particularly inspired by the automatic, continuous calibration used in UNC's HiBall tracker [14]. Projecting calibration patterns during runtime would enable continuous refinement of the calibration parameters. Such a capability would dramatically increase the convenience and utility of portable acquisition and display systems.

Acknowledgements

We thank Tim Weyrich for many fruitful discussions regarding calibration and depth acquisition, Silvio Boehler for the single-shot depth extraction implementation, and Herman Towles, Jim Mahaney, David Harrison and Christian Spagno for help with projectors and hardware design.

References

[1] M. Gross, S. Würmlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. Van Gool, S. Lang, K. Strehlke, A. Vande Moere, and O. Staadt, "blue-c: A spatially immersive display and 3D video portal for telepresence," in *SIGGRAPH 2003 Conference Proceedings*, ACM SIGGRAPH Annual Conference Series, 2003.

[2] R. Raskar, G. Welch, M. Cutts, A. Lake, L. Stesin, and H. Fuchs, "The office of the future: A unified approach to image-based modeling and spatially immersive displays," *Proceedings of SIGGRAPH 98*, pp. 179–188, July 1998.

[3] A. Sadagic, H. Towles, J. Lanier, H. Fuchs, A. van Dam, K. Daniilidis, J. Mulligan, L. Holden, and B. Zeleznik, "National tele-immersion initiative: Towards compelling tele-immersive collaborative environments." presentation given at Medicine meets Virtual Reality 2001 Conference, Jan. 2001.

[4] J. Mulligan and K. Daniilidis, "View-independent scene acquisition for tele-presence," in *Proceedings of the International Symposium on Augmented Reality*, pp. 105–108, 2000.

[5] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti, "Surround-screen projection-based virtual reality: The design and implementation of the cave," *Proceedings of SIGGRAPH 93*, pp. 135–142, August 1993.

[6] M. Agrawala, A. C. Beers, B. Fröhlich, P. M. Hanrahan, I. McDowall, and M. Bolas, "The two-user responsive workbench: Support for collaboration through independent views of a shared space," in *Proceedings of SIGGRAPH 97*, Computer Graphics Proceedings, Annual Conference Series, (Los Angeles, California), pp. 327–332, ACM SIGGRAPH / Addison Wesley, August 1997. ISBN 0-89791-896-7.

[7] R. Raskar, "ilamps," in *SIGGRAPH 2003 Conference Proceedings*, ACM SIGGRAPH Annual Conference Series, 2003.

[8] A. Majumder and R. Stevens, "Color nonuniformity in projection-based displays: Analysis and solutions," *IEEE Transactions on Visualization and Computer Graphics*, vol. 10, pp. 177–188, mar 2004.

[9] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision (2 volumes)*. Reading, MA: Addison-Wesley, 1992.

[10] D. A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*. Prentice Hall, 2002. ISBN 0-13-085198-1.

[11] J. R. Bitner, G. Ehrlich, and E. M. Reingold, "Efficient generation of the binary reflected gray code and its applications," *Communications of the ACM*, vol. 19, no. 9, pp. 517–521, 1976.

[12] P. Vuylsteke and A. Oosterlinck, "Range image acquisition with a single binary-encoded light pattern," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 148–164, Feb 1990.

[13] J. Foley, A. von Dam, S. Feiner, and J. Hughes, *Computer Graphics: Principles and Practice*. Reading, MA: Addison-Wesley, 1992.

[14] G. Welch and G. Bishop, "SCAAT: Incremental tracking with incomplete information," *Computer Graphics*, vol. 31, no. Annual Conference Series, pp. 333–344, 1997.