Abstract: This paper proposes a novel method for synchronizing a high frame-rate (HFR) camera with an HFR projector, using a visual feedback-based synchronization algorithm for streaming video sequences in real time on a visible-light communication (VLC)-based system. The frame rates of the camera and projector are equal, and their phases are synchronized. A visual feedback-based synchronization algorithm is used to mitigate the complexities and stabilization issues of wire-based triggering for long-distance systems. The HFR projector projects a binary pattern modulated at 3000 fps. The HFR camera system operates at 3000 fps, which can capture and generate a delay signal to be given to the next camera clock cycle so that it matches the phase of the HFR projector. To test the synchronization performance, we used an HFR projector–camera-based VLC system in which the proposed synchronization algorithm provides maximum bandwidth utilization for the high-throughput transmission ability of the system and reduces data redundancy efficiently. The transmitter of the VLC system encodes the input video sequence into gray code, which is projected via high-definition multimedia interface streaming in the form of binary images $590 \times 1060$. At the receiver, a monochrome HFR camera can simultaneously capture and decode 12-bit $512 \times 512$ images in real time and reconstruct a color video sequence at 60 fps. The efficiency of the visual feedback-based synchronization algorithm is evaluated by streaming offline and live video sequences, using a VLC system with single and dual projectors, providing a multiple-projector-based system. The results show that the 3000 fps camera was successfully synchronized with a 3000 fps single-projector and a 1500 fps dual-projector system. It was confirmed that the synchronization algorithm can also be applied to VLC systems, autonomous vehicles, and surveillance applications.

Keywords: visible light communication; real-time video processing; high speed vision; wireless video streaming; projector–camera synchronization

1. Introduction

Due to the increasing demand for radio-frequency-based wireless communications, an alternative communication system using light as a source has emerged, known as visible-light communication (VLC) [1–3]. A light source with frequencies ranging between 400 and 800 THz (750 and 375 nm, respectively) is used by VLC systems for encoded transmissions. A photodiode detector is used to detect the modulated visible light and decode it to extract information. With the use of semiconductor light-emitting diodes (LEDs) as a light source that can modulate light at a high speed that is imperceptible to human vision, a VLC system provides the functionality of both communication and room luminescence [4,5].
VLC systems in indoor scenarios, such as homes, offices, or hospital facilities, can provide high-speed internet in a short distance with little interference from other light sources. If a camera is used as a receiver, the most prominent application of an indoor VLC system is visible light positioning (VLP), which combines image processing and data transmission. Image processing is used to recognize the geometry of the environment and monitor mobile nodes [6–8]. With the development of image sensors for the mobile camera, they can be used as a potential receiver in a VLC system [9–11]. Various LED-based VLC applications include intelligent automotive systems [12,13], indoor navigation systems [14], survey measurement systems [15,16], robot control systems [17–19] and LED camera-based VLC [20,21]. The projector-based VLC (PVLC) was studied for imperceptible data projection along with the image [22,23]. Various image-sensor-based VLC systems were developed to decode information transmitted from light sources (e.g., light-emitting diodes (LEDs), display screens, and projectors). The screen intensities of display monitors and LCD panels were modulated using various encoding techniques, and a receiving camera was used to decode the information [24,25]. In a screen-camera-based VLC system, the data shared on the screen can be hidden from view and do not necessarily depend on what is displayed. However, display screens are limited in their encoding functionality. Conversely, the bandwidth limitations of the display-camera-based VLC system can be overcome by using a high frame-rate (HFR) projector and an HFR camera system, which increases the overall bandwidth of the system. The major issue with this system is synchronization, which limits its application.

With the recent development of high-speed complementary metal-oxide-semiconductor (CMOS) image sensors, high frame-rate image processing is highly effective in recognizing a wide variety of high-speed phenomena in the real world. Various HFR vision systems that can process images at hundreds or thousands of frames per second (fps) were developed [26–30]. Many tracking algorithms (e.g., feature-point tracking [31], cam-shift tracking [32], and optical flow estimation [33,34]) were accelerated by using field-programmable gate arrays (FPGAs) and graphics processing units. Therefore, image sensors are widely used in VLC systems as efficient receivers. Additionally, with the increasing demand for projector-based applications, such as structured, light-based, three-dimensional (3D) sensing [35–38], projection mapping [39–42], and interactive applications for augmented reality, there have been extensive developments in projection technology. There are now HFR projector-based systems that can project binary image patterns at thousands of fps or faster, using digital micro-mirror devices (DMDs) [43,44], proving to be the best candidate for a state-of-the-art VLC source.

Because camera synchronization is crucial to some applications, there are some in which the interpolation and prediction of the frames of unsynchronized cameras are used to synchronize the camera [45]. Alleviation of synchronization between the projector and camera can be achieved by using wired external triggers; however, in some applications, there is a need for synchronization without an external trigger, such as in a VLC system. Many industrial cameras are equipped with external trigger-signal ports in which the long wires may cause unstable synchronization, owing to the distance between the projector and camera, whereas short wires may obtrude the constraints of spatial camera configurations. To mitigate these problems, some wireless synchronization applications have been used to display pattern sequences. Correspondingly, the camera achieves synchronization by adjusting its exposure time [46]. There have been studies in which geometric points were used for synchronization, but these require a sufficient number of corresponding points across images to carry out simultaneous geometric calibration and synchronization [47–50]. In some studies [51,52], existing wired standard buses (e.g., IEEE1394 and Ethernet) were used to send trigger signals to the camera for synchronization. Time synchronization is another method that uses Wi-Fi with which the master subsystem provides connectivity and records video, while the slave subsystem provides low-power event detection via ZigBee connectivity [53]. In this system, a Wi-Fi mesh network is used to transmit video data, and a ZigBee network is used to define the network topology and synchronize multiple
surveillance-camera systems. In [54], synchronization was achieved in two stages. During the first stage, the phones were synchronized to the clock of a leader device, using the network time protocol (NTP); during the second stage, all client phone cameras captured a continuous stream that was phase-shifted to achieve better accuracy. The prominent drawback of wireless time synchronization is the nondeterminism of media access time [55]. Another study involved illumination-based intensity-modulated synchronization, using a 1000-fps camera with a phase-locked loop algorithm [56]. This algorithm is robust to background light and locks-in a high camera frame rate to the LED by adjusting the gain parameter to fit the brightness. However, all systems involving camera clock controls using visual feedback require an LED as the light source, or they must be controlled using a wire triggering device or NTP. The wired synchronization system has higher accuracy, but it is unsuitable for an optical wireless communication system.

This paper is an extended version of our previous journal paper [57] in which we proposed a real-time VLC video-streaming system, using an HFR projector and camera. In this paper, we propose a method for synchronizing the HFR camera and projector to resolve inconsistent estimates. This method efficiently reduces data redundancy, mitigates long-term inconsistency issues, and increases the overall bandwidth of the system. A visual feedback-based control algorithm is used to synchronize the HFR camera with the HFR projector by providing a delay to the camera clock to match the phase of the projector’s frame rate. The algorithm generates a delay using the total brightness of an image captured by the HFR camera when a predefined pattern of white and black image sequences is projected. A detailed explanation of the operation of the visual feedback-based synchronization algorithm is described in Section 2. The real-time video reconstruction using a VLC system is explained in Section 3. Section 4 provides an evaluation of the performance of the synchronized system via experiments in real time at 60 fps, with the HFR projector and camera both working at 3000 fps. Finally, in Section 5, the conclusions are drawn.

2. Visual Feedback-Based Projector-Camera Synchronization

2.1. System Configuration

An HFR projector that can control its projection rate is required for synchronization with an HFR camera for high-speed VLC communication. The DLP LightCrafter 4500 (Texas Instruments, Dallas, TX, USA) projector provides a 4000 fps frame rate with bit-plane projection, using 1- to 8-bit projection, supporting a resolution of 912 × 1140. The DLP LightCrafter 4500 is a DMD projection system, where a pixel is represented with a two-dimensional array of electrically addressable and mechanically tiltable micro-mirrors, which are widely used in consumer electronics [58–60]. The DLP projector reproduces the signal by modulating the exposure time of the mirrors over a specific operating refresh time based on the projected frame-bit planes. This feature helps with projecting data at the pixel level, and it transforms the image to a pixel-wise bit-plane projection for the VLC system.

The projected images are captured by an extended version of the monochrome FASTCAM SA-X2 (Photron, Tokyo, Japan) HFR camera, which enables 10,000 fps (real-time) image processing of a megapixel image, using a global electronic shutter with excellent light sensitivity [29]. Figure 1 presents an overview of the FASTCAM SA-X2 with the embedded external board, which was designed to control the delay signal for the camera clock so that it outputs images having a resolution of 512 × 512 with a 12-bit dynamic range at 3125 fps in real time. For an external trigger source, a function generator, AFG1022 (Tektronix, Beaverton, OR, USA), was used to provide a 3000 Hz square-wave trigger at 3.3 V.
2.2. Visual-Feedback-Based Projector–Camera Synchronization

The synchronization method used in the previous paper consists of software-based synchronization. In software synchronization, the HFR camera frame rate is three times the HFR projector frame rate. For example, if the HFR projector frame rate is 1000 fps, then the HFR camera frame rate is 3000 fps. The HFR camera captures three images of a single 1-bit projected image, out of which only the second image with better image brightness, compared to the other two images, is selected. The second image is sometimes not in phase with the HFR projector, which results in the variation of brightness of an image. As a result, we observe inconsistent estimates, data redundancy, and that the overall bandwidth of the system decreases. To overcome the above limitations of software-based synchronization, phase synchronization of HFR camera with HFR projector is required, which can be achieved with the proposed visual feedback-based projector–camera synchronization system as shown in Figure 2. The HFR projector is triggered using an external trigger 1 source (function generator 1), whereas the HFR camera is triggered using another external trigger 2 source (function generator 2). A predefined binary pattern is projected and captured by the HFR camera to calculate the maximum brightness and generate a delay, using a proportional control-based algorithm. This delay is added to the trigger signal of the HFR camera clock to match the HFR camera and projector phases. As a result, we attain maximum and minimum brightness by adding a delay to the camera clock of the HFR camera system.

Figure 1. Overview of the FASTCAM SA-X2 with the embedded external board.

Figure 2. Concept diagram of visual-feedback-based HFR projector–camera synchronization.
To achieve this, the control logic of the timing controller was implemented on an FPGA on the external board attached to the HFR camera, as shown in Figure 3. The sync timing controller accepts the input delay value $\tau$, calculated using the visual-feedback-based algorithm in the software. The value $\tau$ is then fed to the sync signal generator as $\tau_1$ with limits between $\tau_{\text{min}}$ and $\tau_{\text{max}}$. The sync signal generator accepts the input clock frequency $f$ from the external trigger source, and then adds a delay $\tau_1$ to the input clock to generate the output clock signal as SyncOut. This delayed SyncOut clock signal is given as a SyncIn signal to the HFR camera for triggering image capture, thereby matching the phase of the HFR camera with the HFR projector.

Figure 3. Control logic for visual-feedback-based projector–camera synchronization implemented on an external board.

To calculate the delay value, $\tau$, for synchronization, the projector projects a predefined binary pattern of '10101010' continuously, as shown in Figure 4, at a frame rate of $F_p$ with phase $P_p$, while the camera captures individual frames at a frame rate of $F_c$ with phase $P_c$. The white and black images are represented by 1 for maximum brightness and 0 for minimum brightness, respectively. The ideal case where the HFR camera is synchronized with the HFR projector is shown in Figure 4, where the phase $P_c$ is synchronized with phase $P_p$.

Figure 4. Timing diagram of visual-feedback-based projector–camera synchronization.
There are three cases of phase difference as shown in Figure 5. In the first two cases, the phase of the HFR camera is out of phase with the HFR projector (i.e., case-1 and case-2). In case-1, there is a delay in the camera trigger, compared with the projector, whereas in case-2, the camera is triggered before the projector. Thus, projector phase \( P_P \) is not equal to camera phase \( P_C \). In case-3, the camera and projector are both triggered simultaneously, which is the desired result after synchronization.

![Figure 5. Projector–camera synchronization error.](image)

Therefore, case-1 and case-2 can be synchronized, and the HFR camera can be triggered with the same phase as that of the HFR projector by adding a delay of \( \tau \) to \( F_C \), as shown in Figure 6. As with case-1 in Figure 6a, there is a delay in triggering \( F_C \); therefore, to match \( P_C \) with \( P_P \), we must generate a large delay, \( \tau \). In case-2, we require a small delay, \( \tau \), to match \( P_C \) with \( P_P \) as \( F_C \) is triggered prior to the HFR projector, as shown in Figure 6b.

![Figure 6. Timing control of synchronization error: (a) case-1 and (b) case-2.](image)

To calculate the value of \( \tau \), we must initially calculate the brightness-based index, \( R(k) \), which is the ratio of the total brightness of two images at consecutive frames, where
$S(k)$ is the total brightness sum of the input image at the current frame, $k$, and $S(k - 1)$ is the brightness at the previous frame, $k - 1$.

$$R(k) = \frac{S(k)}{S(k - 1)},$$ \hspace{1cm} (1)

$$R(k - 1) = \frac{S(k - 2)}{S(k - 1)}$$ \hspace{1cm} (2)

Next, we evaluate the error, $C(k)$, where $R(k - 1)$ is the previous brightness-based index calculated, using Equation (2).

$$C(k) = R(k) - R(k - 1).$$ \hspace{1cm} (3)

The delay, $\tau$, is calculated for proportional control by considering the delay, $\tau(k - 1)$, at the previous frame and error, $C(k)$, multiplied by a constant proportional gain, $K_{dp}$. The value of $\tau$ lies between $\tau_{\text{max}}$ and $\tau_{\text{min}}$; $\tau_{\text{max}}$ is set to maximum exposure duration of the camera, and $\tau_{\text{min}}$ is zero.

$$\tau(k) = \tau(k - 1) + K_{dp} \cdot C(k) \quad \tau_{\text{max}} > \tau \geq \tau_{\text{min}}.$$ \hspace{1cm} (4)

2.3. Verification

Considering the above-mentioned synchronization algorithm, experiments were conducted to verify its performance at a high frame rate. The HFR projector alternately projected bit-plane images of black and white patterns through a 24-bit color image. The value of each bit plane image (black or white) is decided by the 8-bit pixel value of a single channel, which was set to decimal 170 (10101010 in binary). The HFR projector frame rate was set to 3000 fps, and the exposure of each frame was set to a maximum exposure time of 331 $\mu$s. The HFR camera was set to 3000 fps; the exposure time of each frame was set to $1/3015$ s (331 $\mu$s) to capture the black and white patterns alternately. Because the visual feedback control algorithm depends on the total brightness of an image, the black and white patterns were determined by the maximum and minimum total brightness of the captured image. The output graph of synchronization is shown in Figure 7, where the zoomed portion shows a pattern that the total image brightness drops after every 24 bits, due to the projection of a blank image by the HFR projector between two images. Figure 7 shows that the HFR camera phase, $P_c$, was initially not in sync with the HFR projector phase, $P_p$, and due to this, the blending of projected black and white patterns resulted in inaccurate total brightness of the captured images. Therefore, during the initial duration of approximately 6 s, the total brightness information was inaccurate. After this, the synchronization algorithm was executed, and the value of delay, $\tau(k)$, was calculated using $S(k)$, $R(k)$, and $C(k)$. The delay, $\tau(k)$, was given as input to $\tau_1$ of the sync timing controller on the FPGA of the external board, which generated a $\text{SyncOut}$ signal to trigger the HFR camera and match the phase ($P_c$) with the phase ($P_p$) of HFR projector. The proportional gain, $K_{dp}$, was set to 0.01 to achieve synchronization in shorter duration with stability. Figure 7 shows that the calculated delay, $\tau(k)$, was 0.177 ms, and the HFR camera–projector system was synchronized in approximately 20 ms. As a result, in the synchronized system, the total brightness of the captured image from the HFR camera increased when a white image was projected, and decreased when a black image was projected.
Figure 7. Relationship between total brightness and delay during alternative black-and-white projection at 3000 fps.

HFR camera exposure times of 1/3015, 1/8000, and 1/12,500 were selected to evaluate the performance of our proposed algorithm, which can work with a very short exposure time of 1/12,500 and large exposure time of 1/3015, as shown in Figure 8. The projector frame rate was set to 3000 fps with the exposure time 331 µs, and the bit-plane images of the black and white patterns were projected alternately. The HFR camera frame rate was set to 3000 fps, and the exposure time was kept at 1/3015, 1/8000, and 1/12,500 under a constant room luminescence of 150 lx. Initially, the HFR camera and projector were not in sync for approximately 3 s, after which the synchronization algorithm was executed to calculate the delay, $\tau(k)$, depending on the total brightness of the image. Figure 8 shows that the HFR camera–projector was synchronized under different frame exposure times with different levels of total image brightness. The value of delay $\tau(k)$ was never constant for a particular frame exposure time, and it varied depending on either case-1 and case-2 as discussed above. From this experiment, we can deduce that the system works with a wider range of exposure times and provides system flexibility. Figure 8 shows that synchronization is achievable for all selected ranges, but the only difference is the total brightness of the image.
Figure 8. Relationship between total brightness and delay when 3000 fps black-and-white projection is captured at different exposures.

3. Real-Time Video Streaming Using VLC System

In our previous study, we used real-time video streaming with the VLC system [57], which comprised an HFR projector (DLP LightCrafter 4500), HFR camera (monochrome Fastcam SA-X2 with an additional external board with FPGA), and personal computer (PC) as shown in Figure 9 with its concept. Figure 10 presents a block diagram of the transmitter and receiver section in detail. The transmitter section consists of a gray-coded color video sequence and header information that is fed to the HFR projector for bit-plane binary projection. The receiver section consists of a monochrome HFR camera that captures monochrome images sequentially, from which background subtraction is performed to achieve better thresholding of the bit plane images by combining them. The combined color image is a gray code image, which is further decoded to pure binary code to reconstruct the original image. With reference to our previous work, we added additional information to the header and introduced a method of background subtraction for each image.

Figure 9. Visual-feedback-based projector–camera system.
3.1. Transmitter

The transmitter encoding system involves three stages, as shown in Figure 11, where the input image, \( I_t(x, y) \), is first encoded from pure binary into gray code as \( I_{\text{gray}}(x, y) \), to which header information, \( I_h(w, y) \), is added. Then, the encoded image, \( I_{\text{gray, rgb}}(m, n) \), as shown in Equation (5), is fed to the HFR projector for bit-plane projection.

\[
I_h(w, y) + I_{\text{gray}}(x, y) = I_{\text{gray, rgb}}(m, n),
\]  

(5)

The header information is added to inform the receiver that it received the transmitted information, which consists of five blocks of pixels representing information about the current image as shown in Figure 12. The first block, \( S_0 \), contains all pixel values set to a maximum value of 255 for an 8-bit pixel, which is used to determine the start of a new image. The next five blocks of pixels (i.e., \( F_4, F_3, F_2, F_1, \) and \( F_0 \)) in the header represent a 5-bit frame number, ranging from 0 to 31. Subsequently, two blocks of pixels contain 2-bit channel information \( C_1 \) and \( C_0 \) blocks to represent the red–green–blue (RGB) channels of an image. The next three blocks of pixels, represented as \( B_2, B_1, \) and \( B_0 \), contain 3-bit information of 8-bit planes of a single channel. The \( C_1, C_0, B_2, B_1, \) and \( B_0 \) blocks of pixels aid in determining the sequence of binary images for reconstruction, whereas the last block of pixel \( I_0 \) represents the stream of video, webcam, or video sequence from two different PCs as input streams.
After combining the gray-code image with the respective header information, the image is fed to the HFR projector, where the spatio-temporal projection of an HFR projector is achieved by decomposing a given packed 24-bit $I_{\text{gray_rgb}}(m, n)$ image into its equivalent twenty four 1-bit binary images. The projection pattern used in this system should have the total duration of exposure for all patterns less than or equal to the vsync duration in which a blank sequence is introduced to complete the vsync exposure time.

3.2. Receiver

The projected bit-plane images were captured by a monochrome HFR camera to reconstruct the transmitted 24-bit RGB image. The operation of the receiver section is shown in detail in Figure 13, where the captured images, $C_{\text{in}}(u, v)$, were collected sequentially according to the header information and reconstructed into a gray-code image, $C_{\text{RGB}}(u, v)$, which was converted back to pure-binary code, $I_{\text{RGB}}(u, v)$, to retrieve the transmitted image, $I_t(x, y)$.

The HFR camera was synchronized with an HFR projector, using a visual-feedback-based control algorithm. To retrieve the bit information accurately for each projected bit-plane, a background subtraction method was used to eliminate the noise introduced by ambient light on the projector screen in an indoor office environment. The thresholding method was used for background subtraction, where the reference image was subtracted from the input image. The reference image was estimated using the global thresholding method by projecting the maximum and minimum intensities through the HFR projector onto the screen. The threshold value, $thr(m, n)$, at $(m, n)$ was calculated using Equation (6), where $B(m, n)$ is the pixel value at $(m, n)$ of $C_{\text{in}}(u, v)$, captured after projecting its maximum brightness, and $D(m, n)$ is the pixel value at $(m, n)$ of $C_{\text{in}}(u, v)$, captured after projecting a black image.

![Figure 12. Header information.](image)

![Figure 13. Receiver.](image)
The robustness offered by the background subtraction method is discussed in our previous work. We also introduced an additional background subtraction method in which the reference image is updated for each channel to maximize the efficiency of background subtraction. The bit plane decomposition of an 8-bit image is represented as eight 1-bit binary images with higher bit planes, containing more significant visual information, and the lower bit-plane containing more details. However, the intensity of the lowest bit hardly changes; therefore, we used the lowest bit of each channel to update the reference image, as shown in Figure 14.

![Figure 14. Background subtraction method.](image_url)

Let the 1-bit image of the projected bit-plane image captured by the HFR camera be $C_{in}(u,v)$, and the reconstructed 8-bit image of three channels be combined to form a single 24-bit RGB color image, $C_{RGB}(u,v)$. The $C_{RGB}(u,v)$ image is an encoded gray-code image that is further decoded to a pure-binary-code-based image at the pixel level to obtain the reconstructed RGB color image, $I_{RGB}(u,v)$.

### 3.3. Evaluation Parameter of Image Quality

To assess the image quality of the reconstructed images, a full-reference metrics-based objective image quality index, such as the peak signal-to-noise ratio (PSNR) and multiscale structural similarity index (MS-SSIM) [61] were used. PSNR compares images using different dynamic ranges, as expressed in Equation (7), where MSE is the mean-squared error, and $MAX_i$ is the dynamic range of the allowable pixel intensities. The value of the PSNR should be higher for better image quality.

$$
PSNR = 10 \cdot \log_{10} \left( \frac{MAX_{i}^{2}}{MSE} \right).
$$

The MS-SSIM, as shown in Equation (8), extracts the structural information from the field of view based on the human visual system (HVS) assumption. However, it is not very useful for blurred images, and the measured error lies between zero and one, where one represents the best image quality.

$$
MS-SSIM(x,y) = [l_M(x,y)]^{\alpha_M} \cdot \prod_{j=1}^{M} [c_j(x,y)]^{\beta_j} \cdot [s_j(x,y)]^{\gamma_j}.
$$

To evaluate the efficiency of the reconstructed images, a 5-bit frame number in the header was used by assigning the frame number to each input frame ranging from 1 to 32, thereby creating a packet of 32 frames. The loss is calculated by counting the missing frames within the 32 frames at the receiver. The efficiency of the system was calculated using Equation (9), where $F_r$ is the frame reconstruction efficiency, and $S_r$ represents the
successful frame reconstructed out of the total number of frames, \( F_t \), within one packet of 32 frames. Thus, image quality metrics evaluate the quality of the reconstructed images at the receiver. Additionally, the frame reconstruction efficiency explains the number of frames being reconstructed at the receiver and those being lost because of the bandwidth of the system and luminescence of the HFR projector.

\[
F_t [\%] = \frac{S_r}{F_t} \times 100 \, .
\]  

4. Experiments

To evaluate the performance of the system, we performed various experiments by streaming the saved video and real-time universal serial bus (USB) camera video and reconstructing it, using the VLC system. The HFR projector streamed \( 590 \times 1080 \) video at 60 fps, which is a combination of \( 590 \times 1060 \) gray-code images and \( 590 \times 20 \) header information. This combined image is projected in a bit plane sequence as shown in Figure 15, where the duration of exposure for each bit-plane pattern is \( 331 \, \mu s \). Therefore, the time required to project one frame is \( 24 \times 331 \, \mu s \), which is approximately \( 8000 \, \mu s \) or 8 ms. Therefore, in 1 s, approximately 125 frames can be projected using our system; however, owing to the limitation of the HFR projector, we can transmit \( 590 \times 1080 \) images at maximum 60 fps. A 50-mm lens was mounted on the HFR camera, which was set to the same frame rate as that of the HFR projector (3000 fps). The experimental setup is shown in Figure 16, where the distance between the HFR projector and screen was 950 mm. The projection display onto the screen was \( 448 \times 415 \) mm. The distance between the HFR camera and screen was 1130 mm to ensure that the overall area of the projected video on the screen was captured by the camera. For the proposed system, a stored video sequence was used for a single projector system, and live video streaming from two USB cameras was used to check the synchronization accuracy of the dual projector system. To check the robustness of the system, the indoor environment was illuminated with luminescence values of 0, 150, and 300 lx, using an external light source. The details of the experimental hardware are specified in Figure 17.

![Figure 15. Bit-plane projection pattern for a single RGB image.](image15.png)

![Figure 16. Overview of the HFR projector–camera system.](image16.png)
| Sr.No. | Equipment          | Make                        | Specifications                                      | Quantity |
|-------|--------------------|-----------------------------|-----------------------------------------------------|----------|
| 1     | Receiver PC        | Intel Core i7 @3.00GHz      | RAM : 64GB  Windows 7 Enterprise Microsoft Visual 2015 community | 1        |
| 2     | Transmitter PC-1   | Intel Core i3 @1.90GHz      | RAM : 8GB  Windows 10 Home Microsoft Visual 2015 community | 1        |
| 3     | Transmitter PC-2   | Intel Core i3 @1.90GHz      | RAM : 8GB  Windows 10 Home Microsoft Visual 2015 community | 1        |
| 4     | HFR Camera         | Photron SAX-2 monochrome    | 512x513 @ 3000 fps                                  | 1        |
| 5     | HFR Projector      | DLP lightcrafter 4500       | Input:912x1140 @60fps, output: 1-bit @ 3000/1500 fps | 2        |
| 6     | USB Camera         | Ximea MQ003CG-CM            | 640x480 @ 60 fps                                    | 2        |

Figure 17. Experimental hardware with their specifications.

4.1. Synchronized Real-Time Video Reconstruction

To evaluate the operation of the VLC system after successful synchronization using the visual feedback algorithm, an experiment was performed in which a stored video sequence was streamed in real time. The selected video sequence was from the movie, Big Buck Bunny [62]. Initially, the pure-binary-code images of the 24-bit 1920 × 1080 RGB-color video sequence were gray coded and resized to 590 × 1060 alongside the addition of the 590 × 20 header information. The encoded resized image was projected in bit-plane or binary image sequences at 3000 fps, and the HFR camera captured 512 × 512 images to reconstruct the output image with a resolution of 510 × 459 by combining all bit planes of the 24-bit RGB image sequentially. Figure 18a shows a high definition input image of 1920 × 1080 at 60 fps. Figure 18b contains the reconstructed images 510 × 459, using gray code without background subtraction. Figure 18c depicts the reconstructed images 510 × 459, using gray code with background subtraction. From the reconstructed images, we can deduce that there were no artifacts present when gray-code encoding was used.

Figure 18. Reconstructed saved image sequence on a plain background: (a) 1920 × 1080 input image, (b) 510 × 459 gray-code image without background subtraction, and (c) 510 × 459 gray-code image with background subtraction.
Next, an experiment was conducted to measure the image quality analysis and performance of the system by sending the saved video at 60 fps, which was projected at 3000 fps under different on-screen luminescence conditions of 0, 150, and 300 lx; images were captured at 3000 fps at different exposure times, i.e., 1/3015, 1/8000, and 1/12,500. The results of the image-quality analysis of hundreds of reconstructed images with respect to their original images, are shown in Figures 19 and 20. From the graph shown in Figures 19 and 20, the PSNR and MS-SSIM values indicate that the image quality was better when captured at an exposure of 1/8000 than those at 1/12,500 and 1/3015. However, the background subtraction performed at every frame helped improve the image quality at various HFR camera exposure times. Figure 21 shows the performance of the system based on the number of frames reconstructed at the receiver. A duration of 40 s was observed, and the number of frames was monitored, as shown in Figure 21. The frame reconstruction ratios were almost 100% for 0 lx, whereas for 150 and 300 lx, the frame reconstruction was nearly 100% with small losses. The experimental results indicate that the HFR projector and HFR cameras were synchronized; otherwise, the frame reconstruction would not be possible, and we would not be able to reconstruct the video sequence in real time at 60 fps.

**Figure 19.** PSNRs when a stored video sequence is streamed with pure binary-code and gray-code images on patterned background.
Figure 20. MS-SSIMs when a stored video sequence is streamed with pure binary-code and gray-code images on patterned background.

Figure 21. Cont.
4.2. Real-Time Video Reconstruction Using Two HFR Projectors

The experimental setup for the two projectors is shown in Figure 22, where the dual projectors are maintained such that the projection area overlaps. The distance between the screen and both HFR projectors was kept the same at 950 mm, and the HFR camera was set at a distance of 1130 mm with a 50-mm mounted lens. The experiment scene is shown in Figure 22. In this experiment, the input video sequence was streamed from two USB cameras (XIMEA, MQ003CG-CM) in 24-bit color with a resolution of $640 \times 480$ at 60 fps for transmission, and both cameras were connected to two PCs. The experiment scene comprised a person throwing a football on the floor. In this experiment, HFR projector 2 was set to $180^\circ$ out of phase with respect to HFR projector 1, and both were set to 1500 fps. The bit-plane projection of the image sequence was the same as that in Figure 15. The HFR camera was kept at 3000 fps, that is, double the projection frame rate, which captured the images of each projector alternately and reconstructed both videos at 60 fps. In Figure 23, the two HFR projector input image sequences of $640 \times 480$ at 60 fps are shown. Figure 23b depicts the reconstructed $510 \times 459$ images, using gray code without background subtraction. Figure 23c shows the reconstructed images $510 \times 459$, using gray code with background subtraction. The sequences were reconstructed alternately from the image sequence projected, using HFR projectors 1 and 2.
The image quality analysis and performance of the two projector systems were evaluated by projecting a 60 fps video at 3000 fps under different on-screen luminescence conditions, i.e., 0, 150, and 300 lx, and the images were captured at 3000 fps with different exposure times, i.e., 1/3015, 1/8000, and 1/12,500. Figures 24–27 show that the values of PSNRs and MS-SSIMs were similar, but the best result was observed for exposure 1/3015 at 0 lux. However, the values of MS-SSIM are more promising than the PSNR values because they represent the image quality of the system. Figure 28 shows the performance of the frames reconstructed from each projector system, reflecting the number of frames reconstructed at the receiver corresponding to each projector. Figure 28 shows that the live USB streaming led to few losses in the frame during reconstruction but were not very significant, and the system could reconstruct video sequence in real-time at 60 fps. The
results indicate that multiple projector synchronization was possible with the HFR camera, and the overall bandwidth of the system was utilized.

Figure 24. PSNRs when the USB camera video sequence is streamed through HFR projector-1.

Figure 25. Cont.
Figure 25. MS-SSIMs when the USB camera video sequence is streamed through HFR projector-1.

Figure 26. Cont.
Figure 26. PSNRs when the USB camera video sequence is streamed through HFR projector-2.

Figure 27. MS-SSIMs when the USB camera video sequence is streamed through HFR projector-2.
5. Conclusions

In this article, we presented a novel HFR projector–camera synchronization algorithm using a visual feedback algorithm and evaluated its performance by streaming real-time video using the HFR projector–camera-based VLC system. The experimental results show that synchronization can be achieved at a high frame rate, and that the system is robust to ambient light and can work on a wide range of exposure times. The background subtraction method increased the image quality of the reconstructed image under different ambient light conditions. The experiments were conducted for real-time video streaming to evaluate the percentage of frames received at different frames-per-second and luxes, and it was observed that the frame loss was slightly increased with an increase in the frame rate and lux. The HFR camera and HFR projector system bandwidth were not fully utilized at 3000 fps when a single projector system was used because the system could reconstruct a 60 fps streaming video at nearly 60 fps. Therefore, the dual projector system proved promising, and a full bandwidth of approximately 120 fps was utilized as the dual projector system distributed the computational load of one PC to two. Overall, the images reconstructed using the dual projector had better quality, and the system can be expanded to multiple projectors. The only constraint of the dual projector system is that HFR projector 2 should be triggered by HFR projector 1.

Author Contributions: All authors contributed to the study design and manuscript preparation. I.I. contributed to the concept of HFR-vision-feedback-based synchronization with HFR projector for visible light communication. S.R., K.S. and T.S. designed the high-speed camera-projector system for visible light communication. A.S. developed an algorithm for visual-feedback-based synchronization of an HFR projector–camera system and evaluated its performance using an HFR projector–camera-based visible-light communication system for real-time video streaming. All authors have read and agreed to the published version of the manuscript.
Funding: This research received no external funding.

Data Availability Statement: www.bigbuckbunny.org (accessed on 6 July 2020).

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

References

1. Cevik, T.; Yilmaz, S. An overview of visible light communication systems. *IJCNN 2015*, 7, 139–150. [CrossRef]

2. Bhalerao, M.; Sonavane, S.; Kumar, V. A survey of wireless communication using visible light. *Int. J. Adv. Eng. Technol. 2013*, 5, 188–197.

3. Jovicic, A.; Li, J.; Richardson, T. Visible light communication: Opportunities, challenges and the path to market. *IEEE Commun. Mag. 2013*, 51, 26–32. [CrossRef]

4. Haruyama, S.; Yamazato, T. [Tutorial] Visible light communications. In Proceedings of the IEEE International Conference on Communications, Kyoto, Japan, 5–9 June 2011.

5. Yamazato, T.; Takai, I.; Okada, H.; Fujii, T.; Yendo, T.; Arai, S.; Andoh, M.; Harada, T.; Yasutomi, K.; Kagawa, K.; et al. Image sensor based visible light communication for automotive applications. *IEEE Commun. Mag. 2014*, 52, 88–97. [CrossRef]

6. Chaudhary, N.; Alves, L.N.; Ghassemlooy, Z. Current Trends on Visible Light Positioning Techniques. In Proceedings of the 2019 2nd West Asian Colloquium on Optical Wireless Communications (WACOWC), Tehran, Iran, 27–28 April 2019; pp. 100–105.

7. Chaudhary, N.; Younus, O.I.; Alves, L.N.; Ghassemlooy, Z.; Zvanovec, S.; Le-Minh, H. An Indoor Visible Light Positioning System Using Tilted LEDs with High Accuracy. *Sensors 2021*, 21, 920. [CrossRef]

8. Palacios Játiva, P.; Román Cañizares, M.; Azurdia-Meza, C.A.; Zabala-Blanco, D.; Dehghan Firoozabadi, A.; Seguel, F.; Montejo-Sánchez, S.; Soto, I. Interference Mitigation for Visible Light Communications in Underground Mines Using Angle Diversity Receivers. *Sensors 2020*, 20, 367. [CrossRef] [PubMed]

9. Rajagopal, N.; Lazik, P.; Rowe, A. Visible light landmarks for mobile devices. In Proceedings of the 13th International Symposium on Information Processing in Sensor Networks, Berlin, Germany, 15–17 April 2014; pp. 249–260.

10. Boubezari, R.; Le Minh, H.; Bouridane; Pham, A. Data detection for Smartphone visible light communication. In Proceedings of the 9th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP), Manchester, UK, 23–25 July 2014; pp. 1034–1038.

11. Corbellini, G.; Aksit, K.; Schmid, S.; Mangold, S.; Gross, T. Connecting networks of toys and smartphones with visible light communication. *IEEE Commun. Mag. 2014*, 52, 72–78. [CrossRef]

12. Kasahima, T.; Yamazato, T.; Okada, H.; Fujii, T.; Yendo, T.; Arai, S. Interpixel interference cancellation method for road-to-vehicle visible light communication. In Proceedings of the 2013 IEEE 5th International Symposium on Wireless Vehicular Communications (WiVeC), Dresden, Germany, 2–3 June 2013; pp. 1–5.

13. Chinthaka, H.; Premachandra, N.; Yendo, T.; Yasumoto, M.; Kimura, Y. Detection of LED traffic light by image processing for visible light communication system. In Proceedings of the 2009 IEEE Intelligent Vehicles Symposium, Xi’an, China, 3–5 June 2009; pp. 179–184.

14. Nakajima, M.; Haruyama, S. New indoor navigation system for visually impaired people using visible light communication. *EURASIP J. Wirel. Commun. Netw. 2013*, 2013, 1–10. [CrossRef]

15. Uchiyama, H.; Yoshino, M.; Saito, H.; Nakagawa, M.; Haruyama, S.; Kakehashi, T.; Nagamoto, N. Photogrammetric system using visible light communication. In Proceedings of the 34th Annual Conference of IEEE Industrial Electronics (IECON), Orlando, FL, USA, 10–13 November 2008; pp. 1771–1776.

16. Mikami, H.; Kakehashi, T.; Nagamoto, N.; Nakagomi, M.; Takeomi, Y. Practical Applications of 3D Positioning System using Visible Light Communication; Sumitomo Mitsui Construction Co. Ltd.: Tokyo, Japan, 2011; pp. 79–84.

17. Tanaka, T.; Haruyama, S. New position detection method using image sensor and visible light LEDs. In Proceedings of the 2nd International Conference on Machine Vision (ICMV), Dubai, United Arab Emirates, 28–30 December 2009; pp. 150–153.

18. Nakazawa, Y.; Makino, H.; Nishimori, K.; Nakagomi, M.; Komagata, H. Indoor positioning using a high-speed, fish-eye lens-equipped camera in visible light communication. In Proceedings of the 2013 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Montbeliard, France, 2–31 October 2013.

19. Nakazawa, Y.; Makino, H.; Nishimori, K.; Nakagomi, M.; Komagata, H. High-speed, fish-eye lens-equipped camera based indoor positioning using visible light communication. In Proceedings of the 2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Banff, AB, Canada, 13–16 October 2015.

20. Wang, J.; Kang, Z.; Zou, N. Research on indoor visible light communication system employing white LED lightings. In Proceedings of the IET International Conference on Communication Technology and Application (ICCTA 2011), Beijing, China, 14–16 October 2011; pp. 934–937.

21. Eki, T.; Kiravittaya, S. Demonstration of using camera communication based infrared LED for uplink in indoor visible light communication. In Proceedings of the IEEE Sixth International Conference on Communications and Electronics (ICCE), Ha Long, Vietnam, 27–29 July 2016; pp. 71–76.

22. Nitta, T.; Mimura, A.; Harashima, H. Virtual Shadows in Mixed Reality Environment Using Flashlight-Like Devices. *Trans. Virtual Real. Soc. 2002*, 7, 227–237.
23. Nii, H.; Hashimoto, Y.; Sugimoto, M.; Inami, M. Optical interface using LED array projector. *Trans. Virtual Real. Soc.* **2007**, *12*, 109–117.

24. Dai, J.; Chung, R. Embedding imperceptible codes into video projection and applications in robotics. In *Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura, Portugal, 7–12 October 2012; pp. 4399–4404.

25. Zhang, B.; Ren, K.; Xing, G.; Fu, X.; Wang, C. SBVL Compiler: Secure barcode-based visible light communication for smartphones. *IEEE Trans. Mob. Comput.* **2016**, *15*, 432–446. [CrossRef]

26. Watanabe, Y.; Komuro, T.; Ishikawa, M. 955-fps real-time shape measurement of a moving/deforming object using high-speed imaging for numerous-point analysis. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Roma, Italy, 10–14 April 2007; pp. 3192–3197.

27. Ishii, I.; Taniguchi, T.; Sukenobe, R.; Yamamato, K. Development of high-speed and real-time vision platform, H3 vision. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, St. Louis, MO, USA, 10–15 October 2009; pp. 3671–3678.

28. Ishii, I.; Tatebe, T.; Gu, Q.; Moriue, Y.; Takaki, T.; Tajima, K. 2000 fps real-time vision system with high-frame-rate video recording. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, AK, USA, 3–7 May 2010; pp. 1536–1541.

29. Sharma, A.; Shimasaki, K.; Gu, Q.; Chen, J.; Aoyama, T.; Takaki, T.; Ishii, I.; Tamura, K.; Tajima, K. Super high-speed vision platform that can process 1024 × 1024 images in real time at 12,500 fps. In *Proceedings of the IEEE/SICE International Symposium on System Integration*, Sapporo, Japan, 13–15 December 2016; pp. 544–549.

30. Yamazaki, T.; Katayama, H.; Uehara, S.; Nose, A.; Kobayashi, M.; Shida, S.; Odahara, M.; Takamiya, K.; Hisamatsu, Y.; Matsumoto, S.; et al. A 1ms high-speed vision chip with 3D-stacked 140GOPS column-parallel PEs for spatio-temporal image processing. In *Proceedings of the IEEE International Solid-State Circuits Conference* (ISSCC), San Francisco, CA, USA, 5–9 February 2017; pp. 82–83.

31. Gu, Q.; Raut, S.; Okumura, K.; Aoyama, T.; Takaki, T.; Ishii, I. Real-time image mosaicing system using a high-frame-rate video sequence. *J. Robot. Mechat.* **2015**, *27*, 204–215. [CrossRef]

32. Ishii, I.; Tatebe, T.; Gu, Q.; Takaki, T. Color-histogram-based tracking at 2000 fps. *J. Electron. Imaging* **2012**, *21*, 013010. [CrossRef]

33. Ishii, I.; Taniguchi, T.; Yamamoto, K.; Takaki, T. High-frame-rate optical flow system. *IEEE Trans. Circ. Sys. Video Tech.* **2012**, *22*, 105–112. [CrossRef]

34. Gu, Q.; Nakamura, N.; Aoyama, T.; Takaki, T.; Ishii, I. A full-pixel optical flow system using a GPU-based high-frame-rate vision. In *Proceedings of the 2015 Conference on Advances in Robotics*, Goa, India, 2–4 July 2015.

35. Gao, H.; Aoyama, T.; Takaki, T.; Ishii, I. A Self-Projected Light-Section Method for Fast Three-Dimensional Shape Inspection. *Int. J. Optomechatron.* **2012**, *6*, 289–303. [CrossRef]

36. Liu, Y.; Gao, H.; Gu, Q.; Aoyama, T.; Takaki, T.; Ishii, I. High-frame-rate structured light 3-D vision for fast moving objects. *J. Robot. Mechatron.* **2014**, *26*, 311–320. [CrossRef]

37. Li, B.; An, Y.; Cappelleri, D.; Xu, J.; Zhang, S. High-accuracy, high-speed 3D structured light imaging techniques and potential applications to intelligent robotics. *Int. J. Intell. Robot Appl.* **2017**, *1*, 86–103. [CrossRef]

38. Moreno, D.; Calakli, F.; Taubin, G. Unsynchronized structured light. *ACM Trans. Graph.* **2015**, *34*, 1–11. [CrossRef]

39. Chen, J.; Yamamoto, T.; Aoyama, T.; Takaki, T.; Ishii, I. Simultaneous projection mapping using high-frame-rate depth vision. In *Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China, 31 May–7 June 2014; pp. 4506–4511.

40. Watanabe, Y.; Narita, G.; Tatsuno, S.; Yuasa, T.; Sumino, K.; Ishikawa, M. High-speed 8-bit image projector at 1000 fps with 3 ms delay. In *Proceedings of the International Display Workshops (IDW2015)*, Shiga, Japan, 11 December 2015; pp. 1064–1065.

41. Narita, G.; Watanabe, Y.; Ishikawa, M. Dynamic projection mapping onto deforming non-rigid surface using deformable dot cluster marker. *IEEE Trans. Vis. Comput. Graph.* **2017**, *23*, 1235–1248. [CrossRef]

42. Fleischmann, O.; Koch, R. Fast projector-camera calibration for interactive projection mapping. In *Proceedings of the 23rd International Conference on Pattern Recognition (ICPR)*, Cancun, Mexico, 4–8 December 2016; pp. 3798–3803.

43. Hornbeck, L.J. Digital light processing and MEMS: Timely convergence for a bright future. In *Proceedings of the Plenary Session, SPIE Micromachining and Microfabrication’95*, Austin, TX, USA, 24 October 1995.

44. Youse, J.M. Projection display systems based on the Digital Micromirror Device (DMD). In *Proceedings of the SPIE Conference on Microelectronic Structures and Microelectromechanical Devices for Optical Processing and Multimedia Applications*, Austin, TX, USA, 24 October 1995; Volume 2641, pp. 64–75.

45. Fujiyoshi, H.; Shimizu, S.; Nishi, T. Fast 3D Position Measurement with Two Unsynchronized Cameras. In *Proceedings of 2003 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Kobe, Japan, 16–20 July 2003; pp. 1239–1244.

46. El Asmi, C.; Roy, S. Fast Unsynchronized Unstructured Light. In *Proceedings of the 2018 15th Conference on Computer and Robot Vision (CRV)*, Toronto, ON, Canada, 8–10 May 2018; pp. 277–284.

47. Tuytelaars, T.; Gool, L.V. Synchronizing Video Sequences. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, USA, 27 June–2 July 2004; Volume 1, pp. 762–768.

48. Wolf, L.; Zomet, A. Correspondence-Free Synchronization and Reconstruction in a Non-Rigid Scene. In *Proceedings of the Workshop on Vision and Modeling of Dynamic Scenes*, Copenhagen, Denmark, 28–31 May 2002; pp. 1–19.
49. Tresadern, P.; Reid, I. Synchronizing Image Sequences of Non-Rigid Objects. In Proceedings of the British Machine Vision Conference, Norwich, UK, 9–11 September 2003; Volume 2, pp. 629–638.

50. Whitehead, A.; Laganiere, R.; Bose, P. Temporal Synchronization of Video Sequences in Theory and in Practice. In Proceedings of the IEEE Workshop on Motion and Video Computing, Breckenridge, CO, USA, 5–7 January 2005; pp. 132–137.

51. Rai, P.K.; Tiwari, K.; Guha, P.; Mukerjee, A. A Cost-effective Multiple Camera Vision System Using FireWire Cameras and Software Synchronization. In Proceedings of the 10th International Conference on High Performance Computing, Hyderabad, India, 17–20 December 2003.

52. Litos, G.; Zabulis, X.; Triantafyllidis, G. Synchronous Image Acquisition based on Network Synchronization. In Proceedings of the Conference on Computer Vision and Pattern Recognition Workshop, Washington, DC, USA, 17 June 2006; p. 167.

53. Cho, H. Time Synchronization for Multi-hop Surveillance Camera Systems. In Theory and Applications of Smart Cameras; KAIST Research Series; Kyung, C.M., Ed.; Springer: Dordrecht, The Netherlands, 21 July 2015.

54. Ansari, S.; Wadhwa, N.; Garg, R.; Chen, J. Wireless Software Synchronization of Multiple Distributed Cameras. In Proceedings of the 2019 IEEE International Conference on Computational Photography (ICCP), Tokyo, Japan, 15–17 May 2019; pp. 1–9.

55. Sivrikaya, F.; Yener, B. Time synchronization in sensor networks: A survey. IEEE Netw. 2004, 18, 45–50. [CrossRef]

56. Hou, L.; Kagami, S.; Hashimoto, K. Illumination-Based Synchronization of High-Speed Vision Sensors. Sensors 2010, 10, 5530–5547. [CrossRef] [PubMed]

57. Sharma, A.; Raut, S.; Shimasaki, K.; Senoo, T.; Ishii, I. HFR Projector Camera Based Visible Light Communication System for Real-Time Video Streaming. Sensors 2020, 20, 5368. [CrossRef] [PubMed]

58. Hornbeck, L.J. Digital light processing: A new MEMS-based display technology. In Proceedings of the Technical Digest of the IEEJ 14th Sensor Symposium, Kawasaki, Japan, 4–5 June 1996; pp. 297–304.

59. Gove, R.J. DMD display systems: The impact of an all-digital display. In Proceedings of the Information Display International Symposium, Austin, TX, USA, 13 September 1994; pp. 1–12.

60. Hornbeck, L.J. Digital light processing and MEMS: An overview. In Proceedings of the Digest IEEE/Leeos 1996 Summer Topical Meeting, Advanced Applications of Lasers in Materials and Processing, Keystone, CO, USA, 5–9 August 1996; pp. 7–8.

61. Wang, Z.; Bovik, A.C.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. IEEE Trans. Image Process. 2004, 13, 600–612. [CrossRef]

62. Blender Foundation. 2008. Available online: www.bigbuckbunny.org (accessed on 6 July 2020).