Journal of Visualized Experiments

Robotized Testing of Camera Positions to Determine Ideal Configuration for Stereo 3D Visualization of Open-Heart Surgery

--Manuscript Draft--

Article Type:	Methods Article - JoVE Produced Video
Manuscript Number:	JoVE62786R2
Full Title:	Robotized Testing of Camera Positions to Determine Ideal Configuration for Stereo 3D Visualization of Open-Heart Surgery
Corresponding Author:	Maj Stenmark, Ph.D. Skanes universitetssjukhus Lund Lund, Skane SWEDEN
Corresponding Author's Institution:	Skanes universitetssjukhus Lund
Corresponding Author E-Mail:	maj.stenmark@skane.se
Order of Authors:	Maj Stenmark, phD
	Edin Omerbašić
	Måns Magnusson
	Sanna Nordberg
	Matilda Dahlström
	Phan-Kiet Tran
Additional Information:	
Question	**Response**
Please specify the section of the submitted manuscript.	Medicine
Please indicate whether this article will be Standard Access or Open Access.	Standard Access ($1400)
Please indicate the **city, state/province, and country** where this article will be **filmed**. Please do not use abbreviations.	Sweden
Please confirm that you have read and agree to the terms and conditions of the author license agreement that applies below:	I agree to the Author License Agreement
Please provide any comments to the journal here.	
Please confirm that you have read and agree to the terms and conditions of the video release that applies below:	I agree to the Video Release

TITLE:

Robotized Testing of Camera Positions to Determine Ideal Configuration for Stereo 3D Visualization of Open-Heart Surgery

AUTHORS AND AFFILIATIONS:

Maj Stenmark[1,2], Edin Omerbašić[1,2], Måns Magnusson[1], Sanna Nordberg[3], Matilda Dahlström[3], Phan-Kiet Tran[1]

[1]Cardiac Surgery Unit, Children's Heart Centre, Skåne University Hospital, Lund, Sweden
[2]Department of Clinical Sciences, Lund University, Lund, Sweden
[3]Student at Lund University

Email addresses of co-authors:

Maj Stenmark	(maj.stenmark@skane.se)
Edin Omerbašić	(Edin.Omerbasic@skane.se)
Måns Magnusson	(exoji2e@gmail.com)
Sanna Nordberg	(bmp15sno@student.lu.se)
Matilda Dahlström	(ma2447da-s@student.lu.se)
Phan-Kiet Tran	(phan_kiet.tran@med.lu.se)

Corresponding author:

Maj Stenmark	(maj.stenmark@skane.se)

KEYWORDS:

stereo Vision, 3D, camera baseline, depth perception, robotics, open-heart surgery.

SUMMARY:

The human depth perception of 3D stereo videos depends on the camera separation, point of convergence, distance to, and familiarity of the object. This paper presents a robotized method for rapid and reliable test data collection during live open-heart surgery to determine the ideal camera configuration.

ABSTRACT:

Stereo 3D video from surgical procedures can be highly valuable for medical education and improve clinical communication. But access to the operating room and the surgical field is restricted. It is a sterile environment, and the physical space is crowded with surgical staff and technical equipment. In this setting, unobscured capture and realistic reproduction of the surgical procedures are difficult. This paper presents a method for rapid and reliable data collection of stereoscopic 3D videos at different camera baseline distances and distances of convergence. To collect test data with minimum interference during surgery, with high precision and repeatability, the cameras were attached to each hand of a dual-arm robot. The robot was ceiling-mounted in the operating room. It was programmed to perform a timed sequence of synchronized camera movements stepping through a range of test positions with baseline distance between 50–240 mm at incremental steps of 10 mm, and at two convergence distances of 1100 mm and 1400

mm. Surgery was paused to allow 40 consecutive 5-s video samples. A total of 10 surgical scenarios were recorded.

INTRODUCTION:

In surgery, 3D visualization can be used for education, diagnoses, pre-operative planning, and post-operative evaluation[1,2]. Realistic depth perception can improve understanding[3–6] of normal and abnormal anatomies. Simple 2D video recordings of surgical procedures are a good start. However, the lack of depth perception can make it hard for the non-surgical colleagues to fully understand the antero-posterior relationships between different anatomical structures and therefore also introduce a risk of misinterpretation of the anatomy[7–10].

The 3D viewing experience is affected by five factors: (1) Camera configuration can either be parallel or toed-in as shown in **Figure 1**, (2) Baseline distance (the separation between the cameras). (3) Distance to the object of interest and other scene characteristics such as the background. (4) Characteristics of viewing devices such as screen size and viewing position[1,11–13]. (5) Individual preferences of the viewers[14,15].

Designing a 3D camera setup begins with the capture of test videos recorded at various camera baseline distances and configurations to be used for subjective or automatic evaluation[16–20]. The camera distance must be constant to the surgical field to capture sharp images. Fixed focus is preferred because autofocus will adjust to focus on hands, instruments, or heads that may come into view. However, this is not easily achievable when the scene of interest is the surgical field. Operating rooms are restricted access areas because these facilities must be kept clean and sterile. Technical equipment, surgeons, and scrub nurses are often clustered closely around the patient to secure a good visual overview and an efficient workflow. To compare and evaluate the effect of camera positions on the 3D viewing experience, one complete test range of camera positions should be recording the same scene because the object characteristics such as shape, size, and color can affect the 3D viewing experience[21].

For the same reason, complete test ranges of camera positions should be repeated on different surgical procedures. The entire sequence of positions must be repeated with high accuracy. In a surgical setting, existing methods that require either manual adjustment of the baseline distance[22] or different camera pairs with fixed baseline distances[23] are not feasible because of both space and time constraints. To address this challenge, this robotized solution was designed.

The data was collected with a dual-arm collaborative industrial robot mounted in the ceiling in the operating room. Cameras were attached to the wrists of the robot and moved along an arc-shaped trajectory with increasing baseline distance, as shown in **Figure 2**.

To demonstrate the approach, 10 test series were recorded from 4 different patients with 4 different congenital heart defects. Scenes were chosen when a pause in surgery was feasible: with the beating hearts just before and after surgical repair. Series were also made when the hearts were arrested. The surgeries were paused for 3 min and 20 s to collect forty 5-ssequences with different camera convergence distances and baseline distances to capture the scene. The

videos were later post-processed, displayed in 3D for the clinical team, who rated how realistic the 3D video was along a scale from 0–5.

The convergence point for toed-in stereo cameras is where the center points of both images meet. The convergence point can, by principle, be placed either in front, within, or behind the object, see **Figure 1A–C**. When the convergence point is in front of the object, the object will be captured and displayed left of the midline for the left camera image and right of the midline for the right camera image (**Figure 1A**). The opposite applies when the convergence point is behind the object (**Figure 1B**). When the convergence point is on the object, the object will also appear in the midline of the camera images (**Figure 1C**), which presumably should yield the most comfortable viewing since no squinting is required to merge the images. To achieve comfortable stereo 3D video, the convergence point must be located on, or slightly behind, the object of interest, else the viewer is required to voluntarily squint outwards (exotropia).

The data was collected using a dual-arm collaborative industrial robot to position the cameras (**Figure 2A–B**). The robot weighs 38 kg without equipment. The robot is intrinsically safe; when it detects an unexpected impact, it stops moving. The robot was programmed to position the 5 Megapixel cameras with C-mount lenses along an arc-shaped trajectory stopping at predetermined baseline distances (**Figure 2C**). The cameras were attached to the robot hands using adaptor plates, as shown in **Figure 3**. Each camera recorded at 25 frames per second. Lenses were set at f-stop 1/8 with focus fixed on the object of interest (approximated geometrical center of the heart). Every image frame had a timestamp which was used to synchronize the two video streams.

Offsets between the robot wrist and the camera were calibrated. This can be achieved by aligning the crosshairs of the camera images, as shown in **Figure 4**. In this setup, the total translational offset from the mounting point on the robot wrist and the center of the camera image sensor was 55.3 mm in the X-direction and 21.2 mm in the Z-direction, displayed in **Figure 5**. The rotational offsets were calibrated at a convergence distance of 1100 mm and a baseline distance of 50 mm and adjusted manually with the joystick on the robot control panel. The robot in this study had a specified accuracy of 0.02 mm in Cartesian space and 0.01 degrees rotational resolution[24]. At a radius of 1100 m, an angle difference of 0.01 degrees offsets the center point 0.2 mm. During the full robot motion from 50–240 mm separation, the crosshair for each camera was within 2 mm from the ideal center of convergence.

The baseline distance was increased stepwise by symmetrical separation of the cameras around the center of the field of view in increments of 10 mm ranging from 50–240 mm (**Figure 2**). The cameras were kept at a standstill for 5 s in each position and moved between the positions at a velocity of 50 mm/s. The convergence point could be adjusted in X and Z directions using a graphical user interface (**Figure 6**). The robot followed accordingly within its working range.

The accuracy of the convergence point was estimated using the uniform triangles and the variable names in **Figure 7A** and **B**. The height *'z'* was calculated from the convergence distance *'R'* with the Pythagorean theorem as

$$z = \sqrt{R^2 - \frac{D^2}{4}}$$

When the real convergence point was closer than the desired point, as shown in **Figure 7A**, the error distance 'f_1' was calculated as

$$\frac{D}{e} = \frac{z - f_1}{f_1} \Leftrightarrow f_1 = \frac{ez}{D + e}$$

Similarly, when the convergence point was distal to the desired point, the error distance 'f_2' was calculated as

$$\frac{D}{e} = \frac{z + f_2}{f_2} \Leftrightarrow f_2 = \frac{ez}{D - e}$$

Here, '*e*' was the maximum separation between the crosshairs, at most 2 mm at maximum baseline separation during calibration (*D* = 240 mm). For *R* = 1100 mm (z = 1093 mm), the error was less than ± 9.2 mm. For *R* = 1400 mm (z = 1395 mm), the error was ± 11.7 mm. That is, the error of the placement of the convergence point was within 1% of the desired. The two test distances of 1100 mm and 1400 mm were therefore well separated.

PROTOCOL:

The experiments were approved by the local Ethics Committee in Lund, Sweden. The participation was voluntary, and the patients' legal guardians provided informed written consent.

1. Robot setup and configuration

NOTE: This experiment used a dual-arm collaborative industrial robot and the standard control panel with a touch display. The robot is controlled with RobotWare 6.10.01 controller software and robot integrated development environment (IDE) RobotStudio 2019.5[25]. Software developed by the authors, including the robot application, recording application, and postprocessing scripts, are available at the GitHub repository[26].

CAUTION: Use protective eyeglasses and reduced speed during setup and testing of the robot program.

1.1 Mount the robot to the ceiling or a table using bolts dimensioned for 100 kg as described on page 25 in the product specification[24], following the manufacturer's specifications. Ensure that the arms can move freely and the line of sight to the field of view is unobscured.

CAUTION: Use lift or safety ropes when mounting the robot in a high position.

1.2 Start the robot by turning the start switch located at the base of the robot. Calibrate the robot by following the procedure described in the operating manual on pages 47–56[25].

1.3 Start the robot IDE on a Windows computer.

1.4 Connect to the physical robot system (operating manual page 140[27]).

1.5 Load the code for the robot program and application libraries for the user interface to the robot:

1.5.1 The robot code for a ceiling-mounted robot is in the folder **Robot/InvertedCode** and for a table mounted robot in **Robot/TableMountedCode**. For each of the files left/Data.mod, left/MainModule.mod, right/Data.mod and right/MainModule.mod:

1.5.2 Create a new program module (see operating manual page 318[27]) with the same name as the file (Data or MainModule) and copy the file content to the new module.

1.5.3 Press on **Apply** in the robot IDE to save the files to the robot.

1.6 Use **File Transfer** (operating manual page 346[27]) to transfer the robot application files TpSViewStereo2.dll, TpsViewStereo2.gtpu.dll, and TpsViewStereo2.pdb located in the FPApp folder to the robot. After this step, the robot IDE will not be further used.

1.7 Press the **Reset** button on the back of the robot touch display (FlexPendant) to reload the graphical interface. The robot application Stereo2 will now be visible under the touch display menu.

1.8 Install the recording application (Liveview) and postprocessing application on an Ubuntu 20.04 computer by running the script install_all_linux.sh, located in the root folder in the Github repository.

1.9 Mount each camera to the robot. The components needed for mounting are displayed in **Figure 3A**.

1.9.1 Mount the lens to the camera.

1.9.2 Mount the camera to the camera adaptor plate with three M2 screws.

1.9.3 Mount the circular mounting plate to the camera adaptor plate with four M6 screws on the opposite side of the camera.

1.10 Repeat steps 1.9.1–1.9.3 for the other camera. The resulting assemblies are mirrored, as shown in **Figure 3B** and **Figure 3C**.

1.11 Mount the adaptor plate to the robot wrist with four M2.5 screws, as shown in **Figure 3D**.

1.11.1 For a ceiling-mounted robot: attach the left camera in **Figure 3C** to the left robot arm as shown in **Figure 2A**.

1.11.2 For a table-mounted robot: attach the left camera in **Figure 3C** to the right robot arm.

1.12 Connect the USB cables to the cameras, as shown in **Figure 3E**, and to the Ubuntu computer.

2. Verify the camera calibration

2.1 On the robot touch display, press the **Menu** button and select **Stereo2** to start the robot application. This will open the main screen, as shown in **Figure 6A**.

2.1.1 On the main screen, press on **Go to start** for 1100 mm in the robot application and wait for the robot to move to the start position.

2.1.2 Remove the protective lens caps from the cameras and connect the USB cables to the Ubuntu computer.

2.1.3 Place a printed calibration grid (CalibrationGrid.png in the repository) 1100 mm from the camera sensors. To facilitate correct identification of the corresponding squares, place a small screw-nut or mark somewhere in the center of the grid.

2.2 Start the recording application on the Ubuntu computer (run the script start.sh located in the liveview folder inside the Github repository). This starts the interface, as shown in **Figure 4**.

2.2.1 Adjust the aperture and focus on the lens with the aperture and focus rings.

2.2.2 In the recording application, check **Crosshair** to visualize the crosshairs.

2.3 In the recording application, ensure that the crosshairs align with the calibration grid in the same position in both camera images, as shown in **Figure 4**. Most likely, some adjustment will be required as follows:

2.3.1 If the crosses do not overlap, press the **Gear** icon (bottom left **Figure 6A**) in the robot application on the robot touch display to open the setting screen, as shown in **Figure 6B**.

2.3.2 Press on **1. Go to Start Pos,** as shown in **Figure 6B**.

2.3.3 Jog the robot with the joystick to adjust the camera position (operating manual page 31[23]).

2.3.4 Update the tool position for each robot arm. Press **3. Update Left Tool** and **4. Update Right Tool** to save the calibration for the left and the right arm, respectively.

2.3.5 Press on the **Back Arrow** icon (top right, **Figure 6B**) to return to the main screen.

2.4 Press on **Run Experiment** (**Figure 6A**) in the robot application and verify that the crosshairs align. Otherwise, repeat steps 2.3–2.3.5.

2.5 Add and test any changes to the distances and/or time at this point. This requires changes in the robot program code and advanced robot programming skills. Change the following variables in the **Data** module in the left task (arm): the desired separation distances in the integer array variable **Distances**, the convergence distances in the integer array **ConvergencePos** and edit the time at each step by editing the variable **Nwaittime** (value in seconds).

CAUTION: Never run an untested robot program during live surgery.

2.6 When the calibration is complete, press on **Raise** to raise the robot arms to the standby position.

2.7 Optionally turn off the robot.

NOTE: The procedure can be paused between any of the steps above.

3. Preparation at the start of the surgery

3.1. Dust the robot.

3.1.1 If the robot was turned off, start it by turning on the **Start** switch located at the base of the robot.

3.2 Start the robot application on the touch display and recording application described in steps 2.1 and 2.2.

3.3 In the recording application, create and then select the folder where to save the video (press **Change Folder**).

3.4 In the robot application: press the gear icon, position the cameras in relation to the patient. Change X and Z direction by pressing +/- for **Hand Distance from Robot** and **Height,** respectively, so that the image captures the surgical field. Perform the positioning in the Y-direction by manually moving the robot or patient.

NOTE: The preparations can be paused between the preparation steps 3.1–3.4.

4. Experiment

CAUTION: All personnel should be informed about the experiment beforehand.

4.1. Pause the surgery.

4.1.1 Inform the OR personnel that the experiment is started.

4.2. Press **Record** in the recording application.

4.3. Press **Run experiment** in the robot application.

4.4. Wait while the program is running; the robot displays “Done” in the robot application on the touch display when finished.

4.5. Stop recording in the recording application by pressing **Quit**.

4.5.1. Inform the OR personnel that the experiment has finished.

4.6. Resume surgery.

NOTE: The experiment cannot be paused during steps 4.1–4.6.

5. Repeat

5.1 Repeat steps 4.1–4.6 to capture another sequence and steps 3.1–3.4 and steps 4.1–4.6 to capture sequences from different surgeries. Capture around ten full sequences.

6. Postprocessing

NOTE: The following steps can be carried out using most video editing software or the provided scripts in the postprocessing folder.

6.1 In this case, debayer the video as it is saved in the RAW format:

6.1.1 Run the script postprocessing/debayer/run.sh to open the debayer application shown in **Figure 8A**.

6.1.2 Press **Browse Input Directory** and select the folder with the RAW video.

6.1.3 Press **Browse Output Directory** and select a folder for the resulting debayered and color-adjusted video files.

6.1.4 Press **Debayer!** and wait until the process is finished – both progress bars are full, as shown in **Figure 8B**.

6.2 Merge the right and left synchronized videos to 3D stereo format[28]:

6.2.1 Run the script postprocessing/merge_tb/run.sh to start the merge application; it opens the graphical user interface shown in **Figure 8C**.

6.2.2 Press **Browse Input Directory** and select the folder with the debayered video files.

6.2.3 Press **Browse Output Directory** and select a folder for the resulting merged 3D stereo file.

6.2.4 Press **Merge!** and wait until the finish screen in **Figure 8D** is shown.

6.3 Use off-the-shelf video editing software such as Premiere Pro to add text labels to each camera distance in the video.

NOTE: In the video, there is a visible shake every time the robot moved, and the camera distance increased. In this experiment, labels A–T were used for the camera distances.

7. Evaluation

7.1 Display the video in top-bottom 3D format with an active 3D projector.

7.2 Viewing experience depends on the viewing angle and the distance to screen; evaluate the video using intended audience and setup.

REPRESENTATIVE RESULTS:

An acceptable evaluation video with the right image placed at the top in top-bottom stereoscopic 3D is shown in **Video1**. A successful sequence should be sharp, focused, and without unsynchronized image frames. Unsynchronized video streams will cause blur, as shown in the file **Video 2**. The convergence point should be centered horizontally, independent of the camera separation, as seen in **Figure 9A**,**B**. When the robot transitions between the positions, there is a small shake in the video, which is to be expected at a transition velocity of 50 mm/s. With the too large separation between the right and left image, the brain cannot fuse the images into one 3D image, see **Figure 9C** and **Video 3**.

The position of the heart in the images should be centered during the entire video, as shown in **Figure 1C**. Several reasons can cause this to fail: (1) The convergence point is too far away from the heart, see **Figure 7**. The camera positions relative to the patient can be modified from the robot application setting screen (**Figure 6B**). (2) The camera tool coordinate system is not properly configured. The robot program will simultaneously move the camera symmetrically in a radial motion around the convergence point (**Figure 2C**) and rotate the cameras around the camera tool coordinate system (**Figure 5**). If the camera adaptor plates (**Figure 3**) are assembled or mounted incorrectly, the default values will not work. Rerun step 2.1–2.4 and ensure that the crosshairs in the recording application (**Figure 6**) point at the same object during the full robot motion. When adjusting the coordinate frames, ensure that the object used for calibration (**Figure 4**) is centered between the cameras; otherwise, the calibration will result in non-symmetrical coordinate frames.

If the colors are incorrect after debayering with the debayering application (**Figure 8**), the captured videos have the wrong debayering format. This requires the user to modify the code for the debayering application or use another debayering tool. Similarly, if the automatic

synchronization between the stereo videos failed, the user should use video editing programs such as Premiere Pro to align the videos.

To analyze the results, the video should be displayed on a 3D projector for the intended audience. The audience can subjectively rate how well the 3D video corresponds to the real-life situation. The labels added in step 6.3 can be used to score different distances.

FIGURE AND TABLE LEGENDS:

Figure 1. Placement of convergence point. Different placement of convergence points relative to the object of interest (grey dot). (**A**) Convergence point in front of the object, (**B**) behind the object, and (**C**) on the object. The midline for each camera image is shown with a dotted line. The surgeon is shown from above, standing between the cameras. At the top, the resulting position of the object in the left and the right camera images are displayed relative to the midline.

Figure 2: Robot motion. The camera separation was increased from (**A**) 50 mm to (**B**) 240 mm with incremental steps of 10 mm. (**C**) The robot moved the cameras radially, always pointing the cameras toward the convergence point - the heart. Here the distance *D* is the distance between the cameras, *R* is the radius 1100 or 1400 mm, and *a* is the angle of the cameras, $sin(a) = D/2R$. The right and left cameras were angled *a* degree in the negative and positive direction, respectively, around the tool Z-axis.

Figure 3: Mounting cameras on the robot. (**A**) Exploded view of the components for one camera: lens, camera sensor, camera adaptor plate, circular mounting plate, robot wrist, and screws. The two assembled camera adaptors are shown from (**B**) the robot side and (**C**) the front. (**D**)Adaptors attached to the robot wrist with four M2.5 screws. (**E**) USB cables connected to the cameras.

Figure 4: Camera calibration with the recording application. A calibration grid and a screw-nut were used to calibrate the camera tool coordinate systems relative to the robot writs. The cameras should be angled so that the nut is in the center of the images.

Figure 5: Camera tool coordinate system. The X-axis (red), Y-axis (green), and Z-axis (blue) of the camera tool coordinate system.

Figure 6: The robot application. (**A**) Display of the main screen on the touch display for running the experiments. (**B**) The setup screen for tool calibration and adjustment of the convergence point.

Figure 7: Error estimation. Convergence error (**A**) above and (**B**) below the desired convergence point. The horizontal baseline distance (D = 240 mm), the distance between the cameras, and the convergence point (R = 1100). The vertical distance between the cameras and the convergence point (z = 1093 mm), the maximum separation between the image center points (crosshairs) (e = 2 mm), the vertical error distance when the real convergence point is above the desired convergence position (f_1 = 9 mm). The vertical error distance when the real convergence point is below the desired convergence position (f_2 = 9.2 mm). Figure not drawn to scale.

Figure 8: Postprocessing applications for debayering and merging. (**A**) Start and (**B**) Finish screens of the debayer application. (**C**) Start and (**D**) Finish screens of the merge application.

Figure 9: Snapshots of finished stereo videos. Only every other pixel row was used from the original images to comply with standard top/bottom 3D stereo format. Upper images are from the right camera and lower from the left camera. (**A**) 3D stereo image with 50 mm baseline distance and the convergence point on the OR-table behind the heart. (**B**) 3D stereo image with 240 mm baseline distance and the convergence point at the OR-table behind the heart. (**C**) 3D stereo image with 240 mm baseline distance and the convergence point 300 mm behind the heart.

Video 1. Stereo 3D video at 1100 mm. The convergence point is on the heart, 1100 mm from the cameras. The video starts with a baseline distance of 50 mm (A) and increases with steps of 10 mm to 240 mm (T).

Video 2. Unsynchronized stereo 3D video. The right and left videos are not synchronized which causes blur when viewed in 3D.

Video 3. Stereo 3D video at 1400 mm. The convergence point is behind the heart, 1400 mm from the cameras. The video

DISCUSSION:

During live surgery, the total time of the experiment used for 3D video data collection was limited to be safe for the patient. If the object is unfocused or overexposed, the data cannot be used. The critical steps are during camera tool calibration and setup (step 2). The camera aperture and focus cannot be changed when the surgery has started; the same lighting conditions and distance should be used during setup and surgery. The camera calibration in steps 2.1–2.4 must be carried out carefully to ensure that the heart is centered in the captured video. To troubleshoot the calibration, the values of the camera tool coordinate system can be verified separately by jogging the robot in the coordinate system (step 2.3.3). It is critical to test the full robot program and cameras together with the recording application before the surgery. The height of the operating table is sometimes adjusted during surgery; the height of the robot cameras can also be modified live in the robot application (step 3.4) to keep the desired distance to the heart. The distances and wait times of the robot program can be modified as described in step 2.5.

One limitation of this technique is that it requires that the surgery is paused; therefore, data collection can only be carried out when it is safe for the patient to pause the surgery. Another limitation is that it requires physical adaptation of the operating room to mount the robot in the ceiling and the programmed robot motion assumes that the robot is centered above the heart. Additionally, the cameras are toed-in instead of parallel, which can cause a keystone effect. The keystone effect can be adjusted in postproduction[29–31].

An array of multiple cameras placed on an arc can be used to collect similar data[23]. The camera

array can capture images simultaneously from all cameras; thus, surgery can be paused for a shorter time. A source of error for a camera array is that the cameras can have a slightly different focus, aperture, and calibration and when videos from different camera pairs are compared, other parameters than the baseline distance can affect the image quality and depth perception. Another drawback with a camera array is that the step size between baseline distances is limited by the physical size of the cameras. For example, the lens used in this study has a diameter of 30 mm, which would equal the minimum possible step size. With the setup presented in the study, step sizes of 10 mm were tested but could be set smaller if necessary. Also, with the array setup, height and convergence distance cannot be dynamically adjusted.

Another alternative is to manually move the cameras to predefined positions[22]. This is not feasible during live heart surgery because it would infringe on critical surgical workspace and time.

This method is applicable to many types of open surgery, including orthopedic, vascular, and general surgery, where optimal baseline and convergence distances are yet to be determined.

This method can also be adapted to collect images for purposes other than 3D visualization. Many computer vision applications use the disparity between images to calculate the distance to an object. A precise camera motion can be used to 3D scan stationary objects from multiple directions to create 3D models. For 3D localization, the 3D viewing experience is less important if the same points on the object can be identified in different images, depending on accurate camera positioning, camera calibration, light conditions, and frame synchronization.

Robot-controlled camera positioning is both safe and effective for collecting video data for the identification of optimal camera positions for stereoscopic 3D video.

ACKNOWLEDGMENTS:
The research was carried out with funding from Vinnova (2017-03728, 2018-05302 and 2018-03651), Heart-Lung Foundation (20180390), Family Kamprad Foundation (20190194), and Anna-Lisa and Sven Eric Lundgren Foundation (2017 and 2018).

DISCLOSURES:
The authors have nothing to disclose.

REFERENCES:
1. Held, R. T., Hui, T. T. A guide to stereoscopic 3D displays in medicine. *Academic Radiology*. **18** (8), 1035–1048 (2011).
2. van Beurden, M. H. P. H., IJsselsteijn, W. A., Juola, J. F. Effectiveness of stereoscopic displays in medicine: A review. *3D Research*. **3** (1), 1–13 (2012).
3. Luursema, J. M., Verwey, W. B., Kommers, P. A. M., Geelkerken R. H., Vos, H. J. Optimizing conditions for computer-assisted anatomical learning. *Interacting with Computers*. **18** (5), 1123–1138 (2006).
4. Takano, M. et al. Usefulness and capability of three-dimensional, full high-definition

movies for surgical education. *Maxillofacial Plastic and Reconstructive Surgery*. **39** (1), 10 (2017).
5. Triepels, C. P. R. et al. Does three-dimensional anatomy improve student understanding? *Clinical Anatomy*. **33** (1), 25–33 (2020).
6. Beermann, J. et al. Three-dimensional visualisation improves understanding of surgical liver anatomy. *Medical Education*. **44** (9), 936–940 (2010).
7. Battulga, B., Konishi, T., Tamura, Y., Moriguchi, H.. The Effectiveness of an interactive 3-dimensional computer graphics model for medical education. *Interactive Journal of Medical Research*. **1** (2) (2012).
8. Yammine, K., Violato, C. A meta-analysis of the educational effectiveness of three-dimensional visualization technologies in teaching anatomy. *Anatomical Sciences Education*. **8** (6), 525–538 (2015).
9. Fitzgerald, J. E. F., White, M. J., Tang, S. W., Maxwell-Armstrong, C. A., James, D. K. Are we teaching sufficient anatomy at medical school? The opinions of newly qualified doctors. *Clinical Anatomy*. **21** (7), 718–724 (2008).
10. Bergman, E. M., Van Der Vleuten, C. P. M., Scherpbier, A. J. J. A. Why don't they know enough about anatomy? A narrative review. *Medical Teacher*. **33** (5), 403–409 (2011).
11. Terzić, K., Hansard, M. Methods for reducing visual discomfort in stereoscopic 3D: A review. *Signal Processing: Image Communication*. **47**, 402–416 (2016).
12. Fan, Z., Weng, Y., Chen, G., Liao, H. 3D interactive surgical visualization system using mobile spatial information acquisition and autostereoscopic display. *Journal of Biomedical Informatics*. **71**, 154–164 (2017).
13. Fan, Z., Zhang, S., Weng, Y., Chen, G., Liao, H. 3D quantitative evaluation system for autostereoscopic display. *Journal of Display Technology*. **12** (10), 1185–1196 (2016).
14. McIntire, J. P. et al. Binocular fusion ranges and stereoacuity predict positional and rotational spatial task performance on a stereoscopic 3D display. *Journal of Display Technology*. **11** (11), 959–966 (2015).
15. Kalia, M., Navab, N., Fels, S. S., Salcudean, T. A method to introduce & evaluate motion parallax with stereo for medical AR/MR. *IEEE Conference on Virtual Reality and 3D User Interfaces*. 1755-1759 (2019).
16. Kytö, M., Hakala, J., Oittinen, P., Häkkinen J. Effect of camera separation on the viewing experience of stereoscopic photographs. *Journal of Electronic Imaging*. **21** (1), 1–9 (2012).
17. Moorthy, A. K., Su, C. C., Mittal, A., Bovik, A. C. Subjective evaluation of stereoscopic image quality. *Signal Processing: Image Communication*. 28(8), 870-883 (2013).
18. Yilmaz, G. N. A depth perception evaluation metric for immersive 3D video services. 3DTV Conference: *The True Vision - Capture, Transmission and Display of 3D Video*. 1–4 (2017).
19. Lebreton, P., Raake, A., Barkowsky, M., Le Callet, P. Evaluating depth perception of 3D stereoscopic videos. *IEEE Journal on Selected Topics in Signal Processing*. **6**, 710–720 (2012).
20. López, J. P., Rodrigo, J. A., Jiménez, D., Menéndez, J. M. Stereoscopic 3D video quality assessment based on depth maps and video motion. *EURASIP Journal on Image and Video Processing*. **2013** (1), 62 (2013).
21. Banks, M. S., Read, J. C., Allison, R. S., Watt, S. J. Stereoscopy and the human visual system. *SMPTE Motion Imaging Journal*. **121** (4), 24–43 (2012).
22. Kytö, M., Nuutinen, M., Oittinen, P. Method for measuring stereo camera depth accuracy based on stereoscopic vision. *Three-Dimensional Imaging, Interaction, and Measurement*. **7864**,

168–176 (2011).
23. Kang, Y. S., Ho, Y. S. Geometrical compensation algorithm of multiview image for arc multi-camera arrays. *Advances in Multimedia Information Processing*. **2008**, 543–552 (2008).
24. *ABB Robotics. Product Specification IRB 14000. DocumentID: 3HAC052982-001 Revision J* at < https://library.abb.com/en/results > (2018).
25. *ABB Robotics. Operating Manual IRB 14000. Document ID: 3HAC052986-001 Revision F* at < https://library.abb.com/en/results > (2019).
26. Githhub repository at <https://github.com/majstenmark/stereo2> (2021).
27. *ABB Robotics. Operating manual RobotStudio. Document ID: 3HAC032104-001 Revision Y* at < https://library.abb.com/en/results > (2019).
28. Won, C. S. Adaptive interpolation for 3D stereoscopic video in frame-compatible top-bottom packing. *IEEE International Conference on Consumer Electronics*. **2011**, 179–180 (2011).
29. Kim, S..K., Lee, C., Kim, K. T. *Multi-view image acquisition and display*. In: Javidi, B., Okano, F., Son, J. Y. (eds). *Three-Dimensional Imaging, Visualization, and Display*. 227–249, Springer, New York (2009).
30. Liu, F., Niu, Y., Jin, H. Keystone correction for stereoscopic cinematography. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. **2012**, 1–7 (2012).
31. Kang, W., Lee, S. Horizontal parallax distortion correction method in toed-in camera with wide-angle lens. *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video*. **2009**, 1–4 (2009).

Figure1

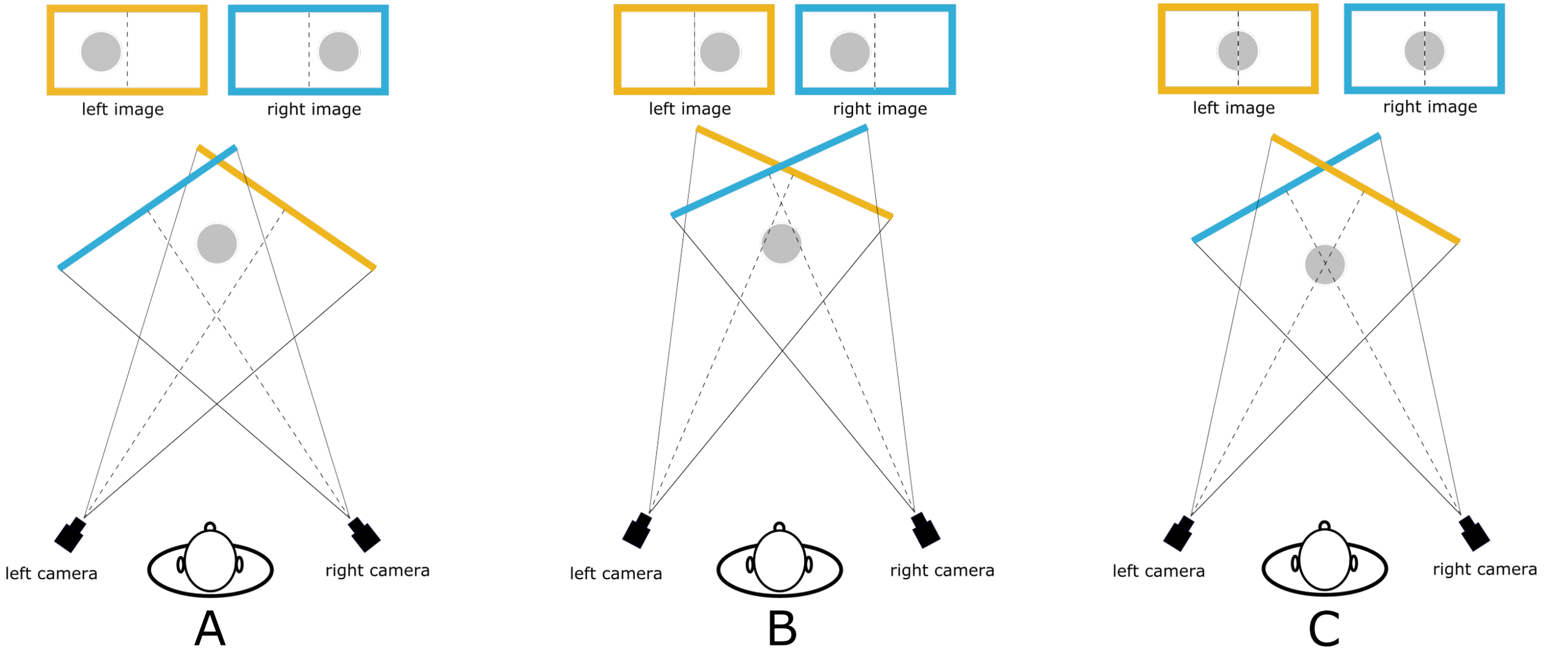
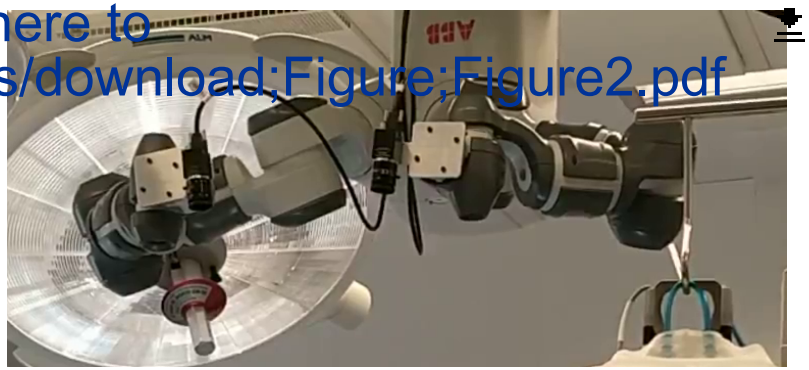
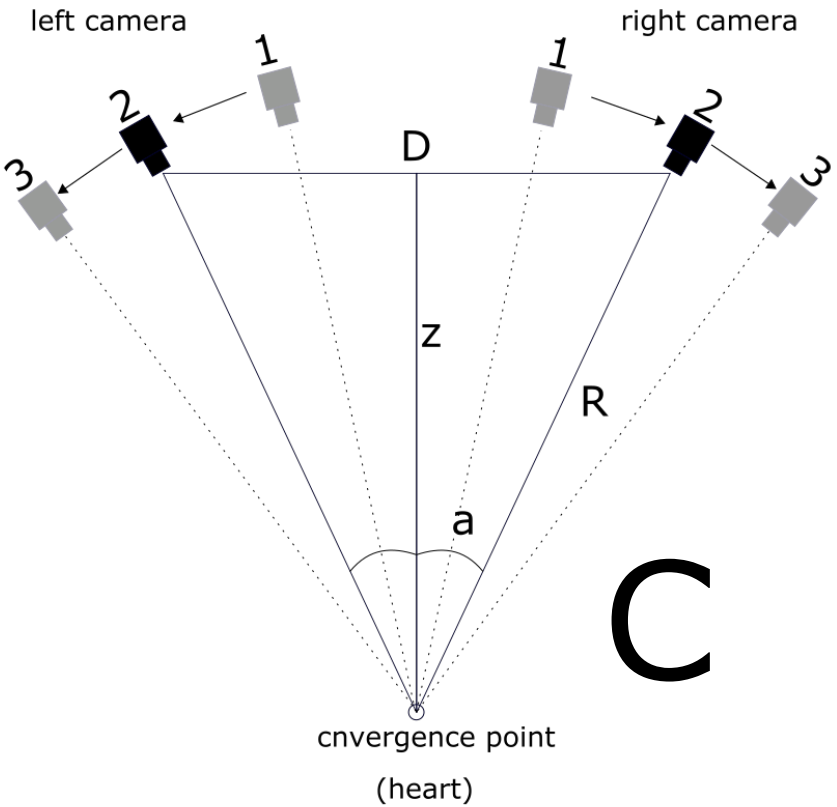


Figure2

A



B



C

Figure3

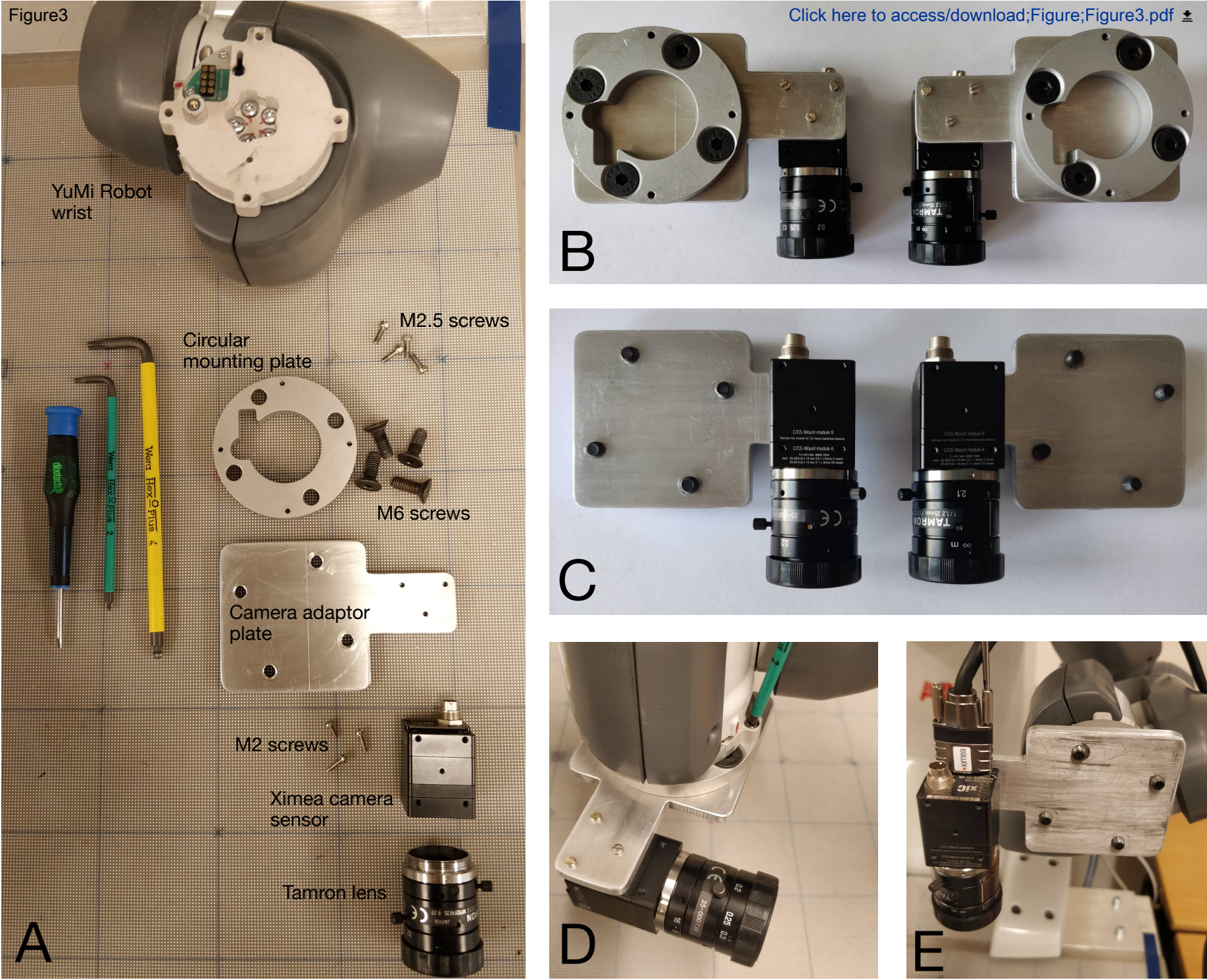


Figure4

Click here to access/download;Figure;Figure4.png

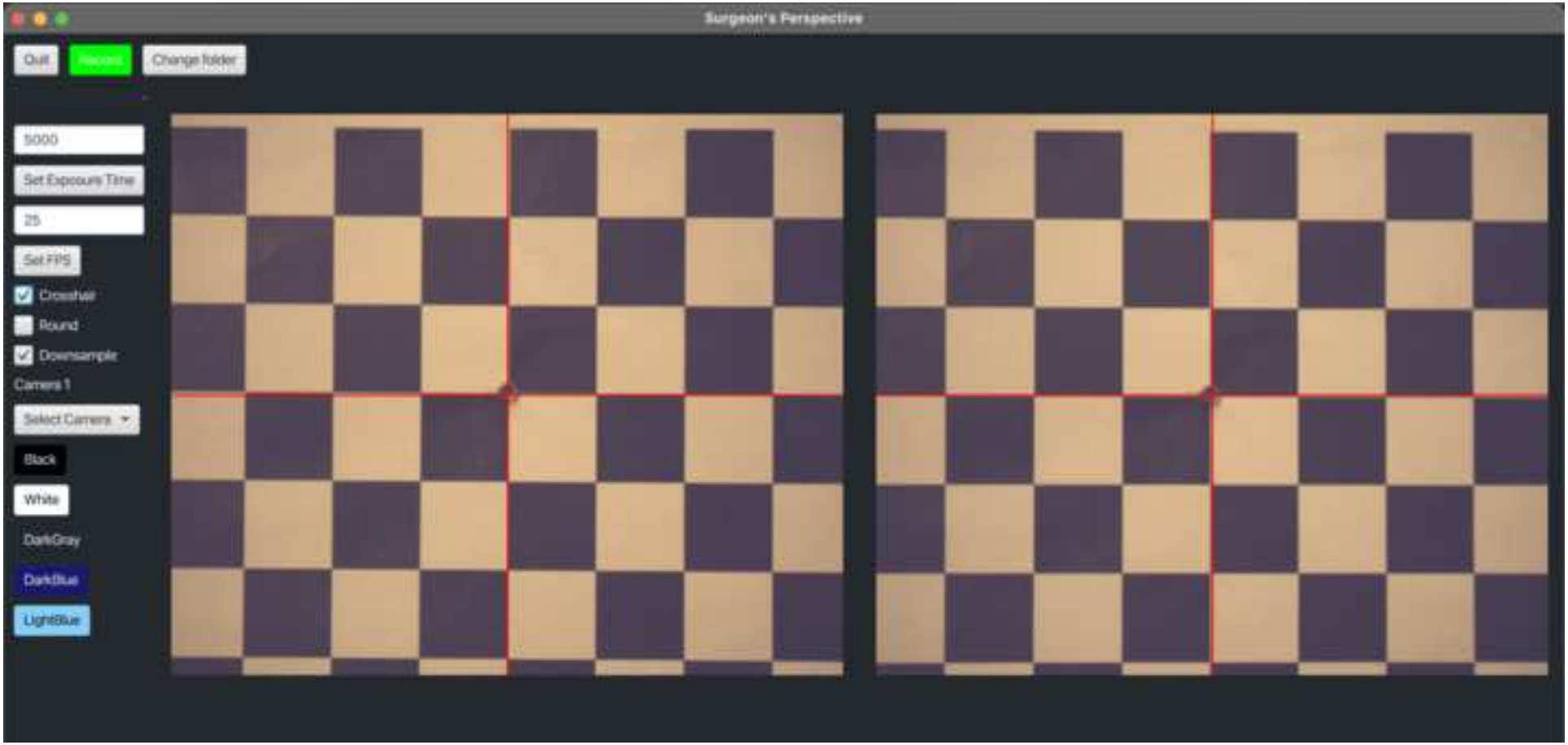


Figure5

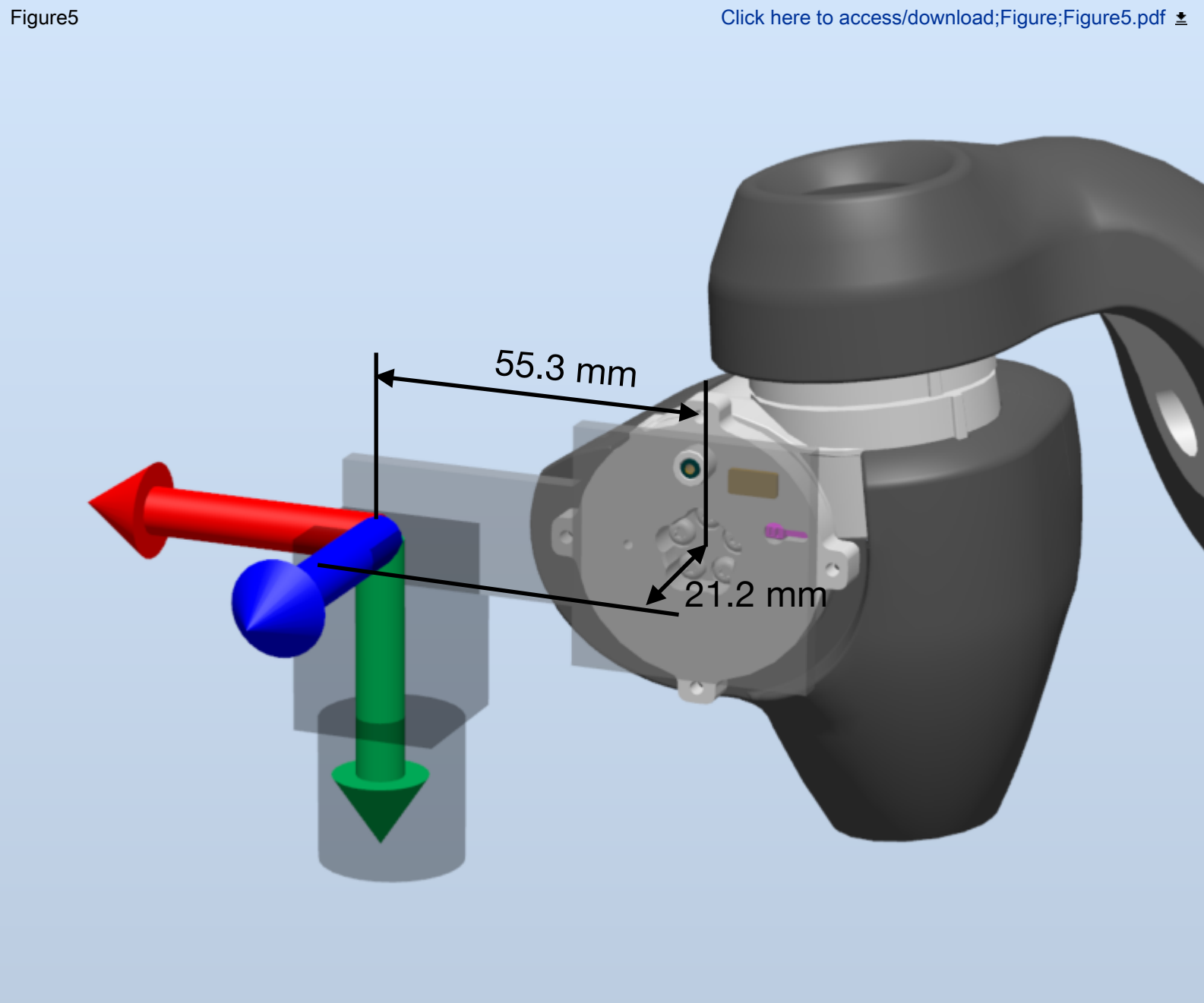
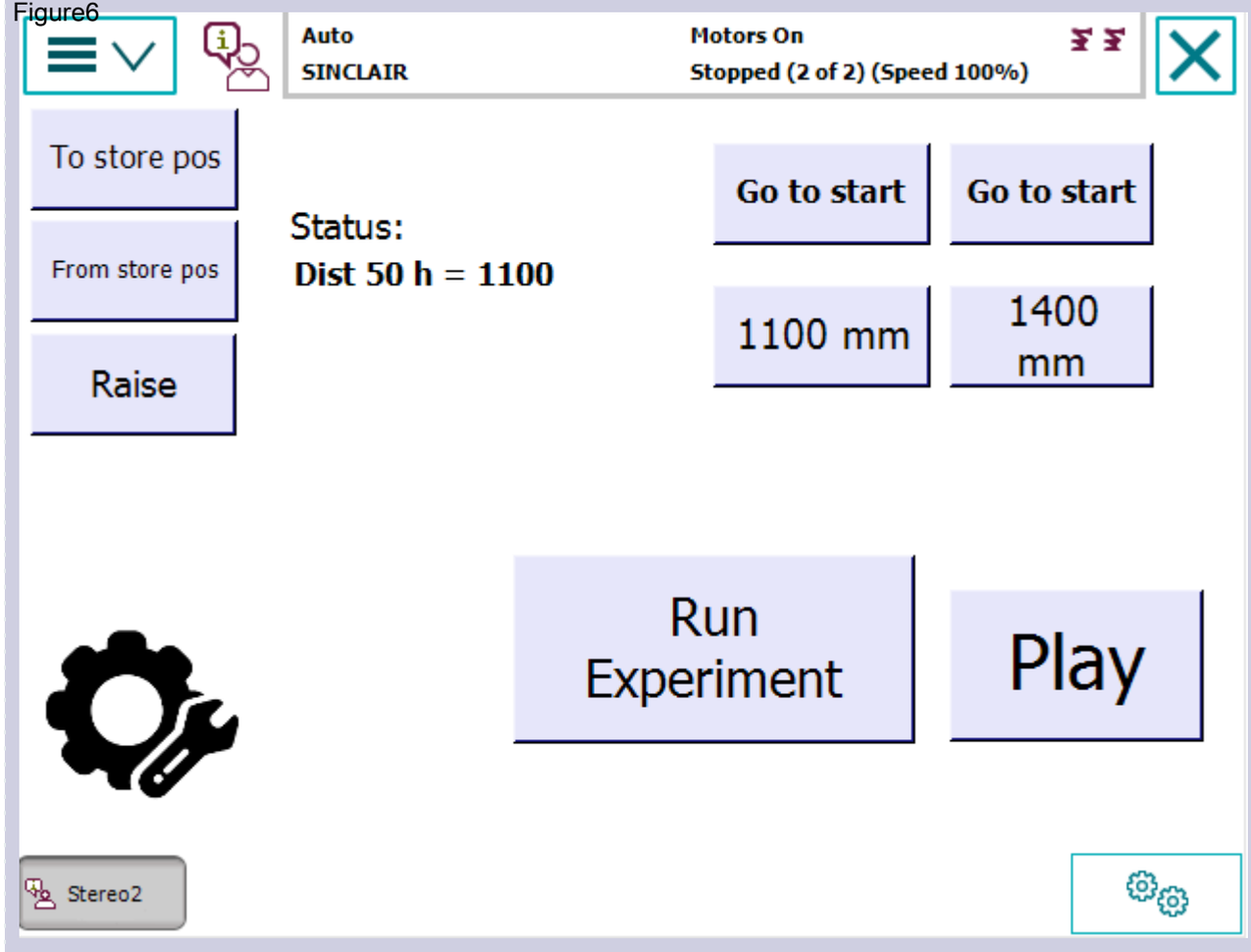
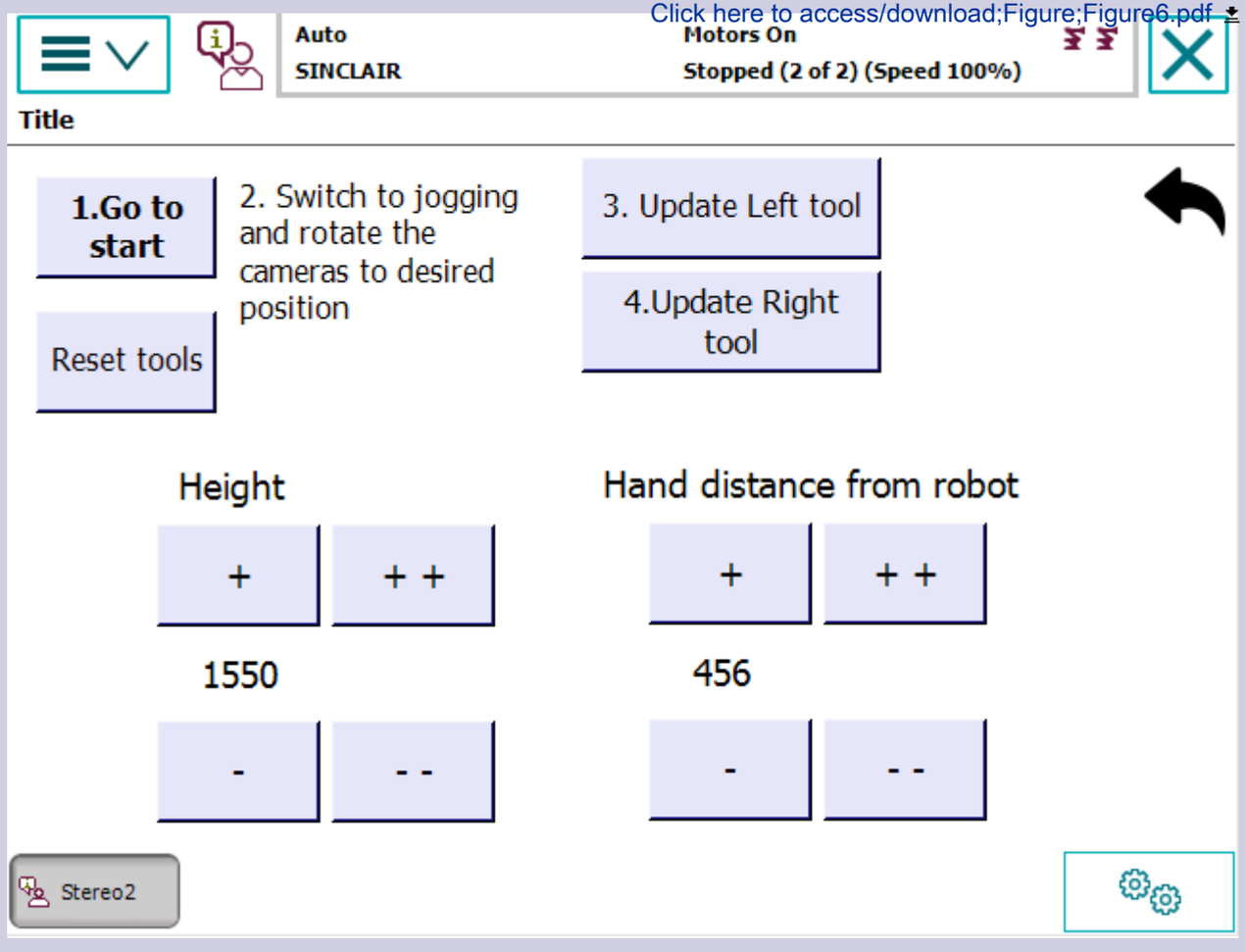


Figure6



A



B

Figure7

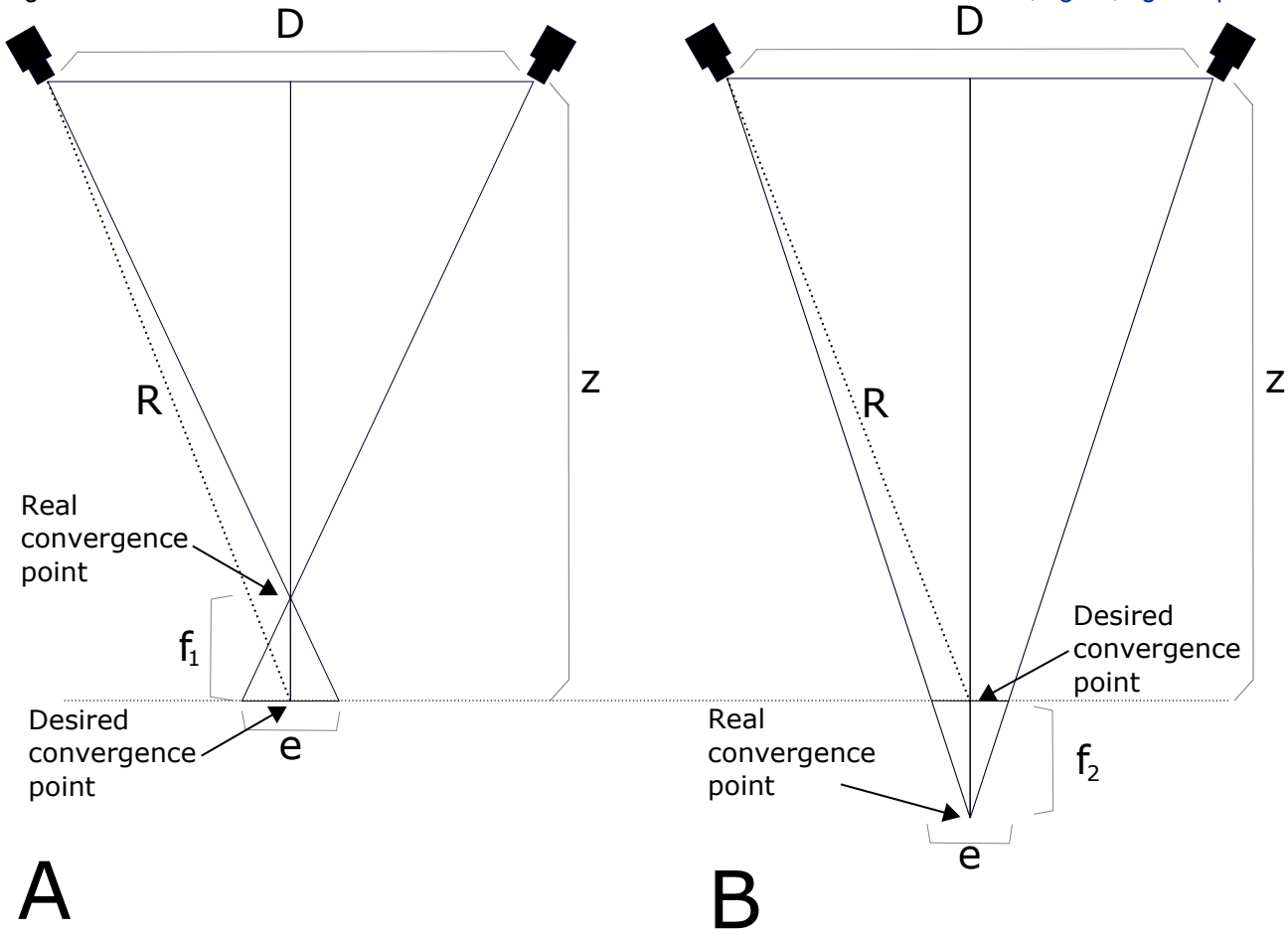


Figure8

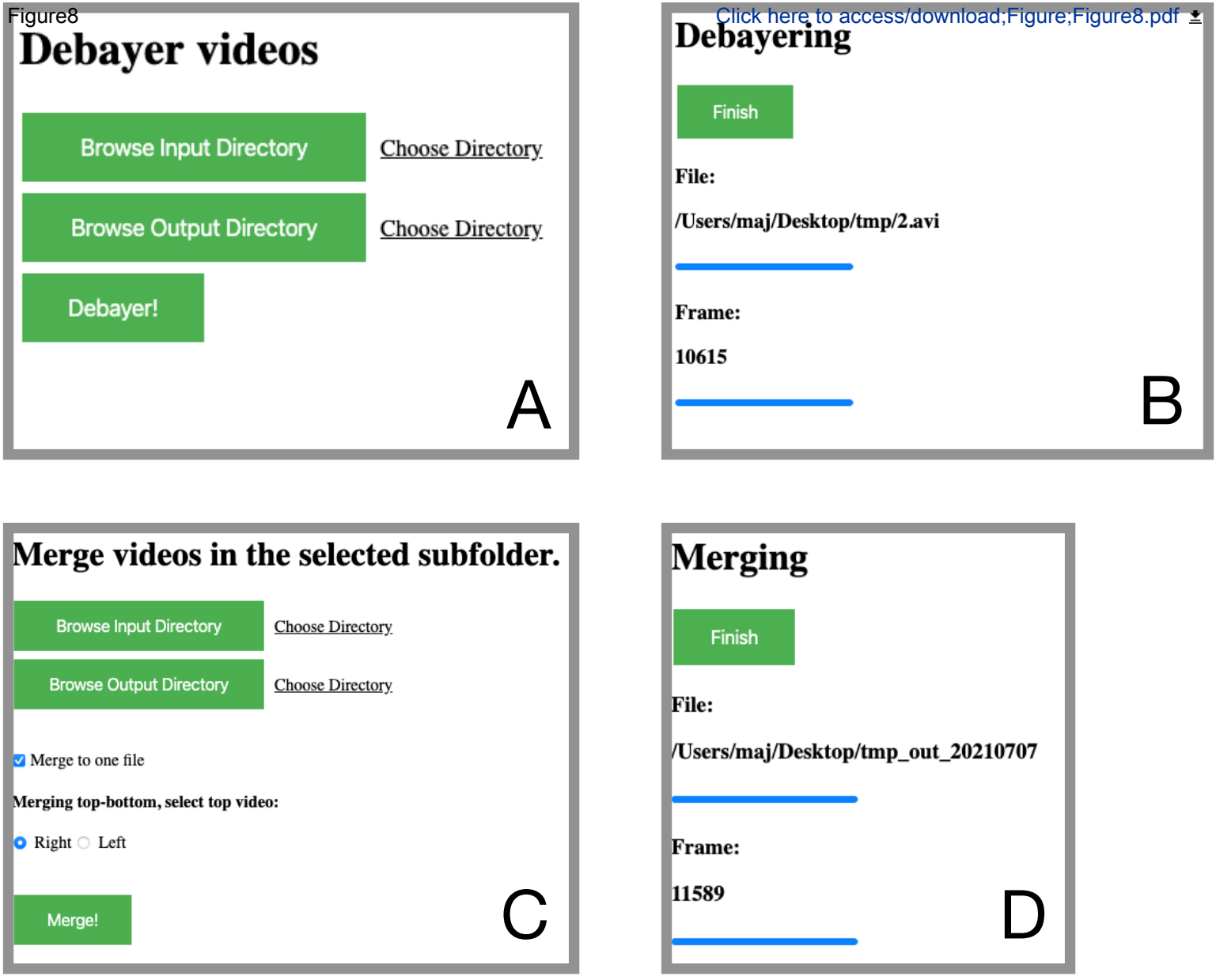
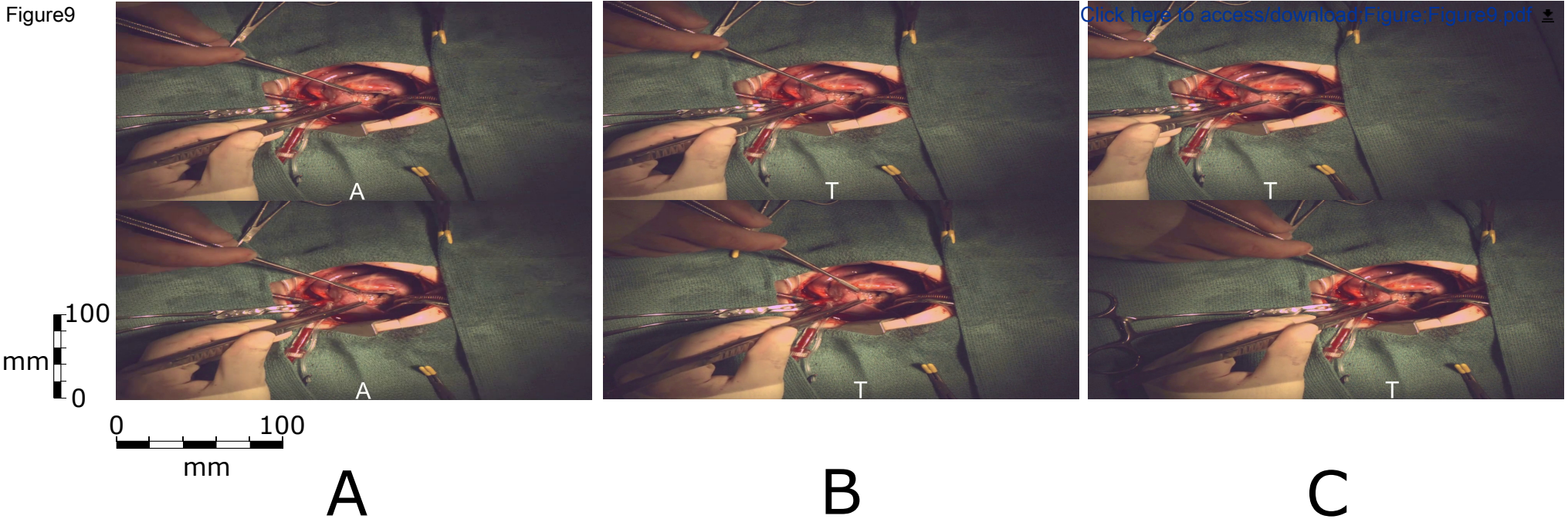


Figure9



Video 1. Stereo 3D video at 1100 mm. The convergence point is on the heart, 1100 mm from the cameras. The video starts with

Video 2. Unsynchronized stereo 3D video. The right and left videos are not synchronized which causes blur when viewed in

Click here to access/download
Video or Animated Figure
Video2.mp4

Video 3. Stereo 3D video at 1400 mm. The convergence point is behind the heart, 1400 mm from the cameras. The video starts

Note by authors: Line numbers refer to the line numbers when the tracked changes markup is set to “No Markup”.

1. The manuscript has been formatted to fit the journal standard. Please check and verify the step numbers mentioned in the steps (e.g., line 209, 256, 281, 291, 318, etc.).

The referenced steps are updated on lines 259, 450, 452. The references on line 208, 283, 293, 316, 320, 382, 396, 458, and 459 are verified.

2. JoVE policy states that the video narrative is objective and not biased towards a particular product featured in the video. The goal of this policy is to focus on the science rather than to present a technique as an advertisement for a specific item. To this end, we ask that you please reduce the number of instances of " RobotWare, RobotStudio, Liveview, Ximea, Tamron, YuMi, ABB, etc." within your text. The term may be introduced, but please use it infrequently and when directly relevant. Otherwise, please refer to the term using generic language. Please reference all the commercial terms in the Table of Materials.

Changes have been made accordlingly. We refer to Liveview with the general term “recording application”, RobotStudio as “robot integrated development environment (IDE)” and Flexpendant as “touch display” on lines 155, 171, 185, 189, 191, 192, 222, 235, 240, 247, 283, 308, 310, 383, 417, 424.

3. Please highlight up to 3 pages of the Protocol (including headings and spacing) that identifies the essential steps of the protocol for the video, i.e., the steps that should be visualized to tell the most cohesive story of the Protocol. Remember that non-highlighted Protocol steps will remain in the manuscript, and therefore will still be available to the reader.

We have highlighted lines 198-244, 288-314 and 360.

4. Please include a title and a description of each video uploaded.

We have added the following titles and descriptions during the uploading process:

Video 1. Stereo 3D video at 1100 mm. The convergence point is on the heart, 1100 mm from the cameras. The video starts with baseline distance 50 mm (A) and increase with steps of 10 mm to 240 mm (T).

Video 2. Unsynchronized stereo 3D video. The right and left videos are not synchronized which causes blur when viewed in 3D.

Video 3. Stereo 3D video at 1400 mm. The convergence point is behind the heart, 1400 mm from the cameras. The video starts with baseline distance 50 mm (A) and increase with steps of 10 mm to 240 mm (T).

5. Please consider including any data on the scoring/evaluation of the video.

The evaluation process should be individualized for each use case. The evaluation methodology is not as simple as it may appear and in itself probably beyond the scope of this manuscript. Our work on the evaluation method has been and is currently being upheld by the

ongoing pandemic and local restrictions. Here is a brief summary to provide the editor an idea of where we are at this point, and why more work needs to be done on the subject:

A total of 18 medical staff members working at the Children's Heart Centre in Lund were asked to provide an overall score of the 3D visual experience of recordings from three different surgical procedures. To avoid potential pattern-recognition-bias, video sequences of different baselines (camera separations) were labelled with letters instead of exact measurements. The testing sequence always started with from the narrowest to the widest baseline, and the viewers were asked to provide a relative score beginning at 3 on a scale from 1 to 5, where 5 is very good, and 1 is very bad.

Convergence point 1400 mm from the cameras, 300 mm below the heart, resulted in tunnel vision and discomfort. The convergence point at the level of the heart, 1100 mm from camera sensor, provided the best 3D representation of ground truth. The scoring varied vastly between individuals. No discrete, statistically significant, peak of best scores could be identified by this preliminary evaluation method.

The discrepancy in scoring may depend on the extend of previous experience and knowledge of ground truth, where large gaps are to be expected between surgeons and all other categories of non-surgical staff including cardiologist, radiologist and anesthesiologist and different level of nursing staff, all of whom participated in the evaluation. In addition, when a group of 20 engineering students were asked to evaluate the 3D videos, two of them were found to have stereo blindness.

The evaluation method must be refined. In depth interview with individual viewer (evaluator) as well as selection of those with adequate abilities and experience to evaluate, whom would probably be the surgeons and surgeons within the specific specialty being recorded only. This work is underway.

6. Figure 7: Please specify the values presented in the lines 130-141 in the figure legends.

The figure has been updated for clarity, with the distance R included (previously z was estimated to be equal to R). Lines 131-144 were updated accordingly. Figure legend was updated with values for the smallest convergence distance (R = 1100).

Adaptor plate

Click here to access/download

Supplemental Coding Files

Robot_camera_adaptor_plates.dwg

Circular adaptor plate

Click here to access/download

Supplemental Coding Files

Mountingplate_ROBOT_SIDE_NewDesign_4.STL