Projecting Trackable Thermal Patterns for Dynamic Computer Vision: Supplemental Materials

Mark Sheinin, Aswin C. Sankaranarayanan, and Srinivasa G. Narasimhan
Carnegie Mellon University, Pittsburgh, PA 15213, USA

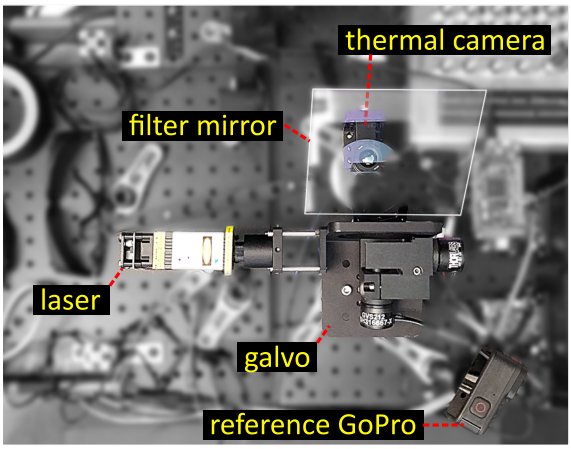

Figure 1. System prototype, top view.

1. Prototype Hardware-related details

1.1. Prototype components description

An image of our system is shown in Fig. 1. The thermal camera is an Industrial Grade FLIR Boson+ 640 having 24° Horizontal Field of View (HFOV) [1]. The projector is a combination of a laser and a Thorlabs GVS212 dual-axis scanning galvo systems. We used two lasers in our experiment: a $532\,\mathrm{nm}$ $150\,\mathrm{mW}$ Coherent Sapphire low-power (LP), and an off-the-shelf $520\,\mathrm{nm}$ $1\,\mathrm{W}$ consumer grade laser [2]. Besides a sufficiently high wattage to quickly create a slight surface temperature rise, our method requires no special laser features (*e.g.*, temporal or spectral stability, beam width, and more). The pattern is executed by sending analog voltages to the galvo systems from an NI USB-6343 DAQ. The projector and camera are temporarily synchronized using separate clocks from an Arduino Due to achieve precise control of the pattern points projected in each frame.

1.2. Microbolometer thermal cameras

In this paper, we image the scene using an LWIR microbolometer-based sensor whose image formation model differs from standard visible-light cameras [11]. Visible-light CMOS and CCD cameras measure the incoming light intensity by 'counting' the number of photons arriving at each pixel during the exposure time. Conversely, a microbolometer sensor measures the incident power of infrared light during continuous exposure. The incident power per pixel is sampled at regular time intervals T^{samp} and converted to temperature readings, where $1/T^{\mathrm{samp}}$ yields capture speeds similar to standard cameras (*e.g.*, $30\,\mathrm{Hz}$). This difference in image formation models benefits our method, as explained next.

In a standard RGB camera, the motion of a scene light source *during the exposure time* results in a 'motion blur' curve. Given the image of the resulting curve, it is generally impossible to discern the temporal light source trajectory during the exposure and, specifically, its location at the end of the exposure time (unless additional light encoding is used [7]). Conversely, roughly speaking, the 'motion blur' model of a microbolometer follows an exponential decay backward in time. This means that for the same heat point source trajectory that resulted in a curve for the RGB camera, our thermal image would result in a point having an exponentially decaying 'trail.' Thus, the point's location at the frame sample time is much easier to determine and roughly corresponds to the point's peak image intensity.

1.3. Projector-camera synchronization constraints

The synchronization between the camera and the galvo was achieved by generating digital sync clocks using an Arduino. The clock periods were determined using Arduino's `delayMicroseconds()` function. Since the function is limited to integer microseconds (*e.g.*, $1666\,\mu\mathrm{sec}$ for a $30\,\mathrm{Hz}$ clock), generating precise clock fractions was impossible (*e.g.*, $1666/3$ for $K = 3$).

1.4. Laser safety considerations

Our prototype used green lasers whose power was sufficiently high such that unintended eye exposure may be harmful. However, our system can borrow the power regime of long-range LIDARs that use a $1550\,\mathrm{nm}$ lasers. At this wavelength, a laser can be 40x times more powerful and still be considered eye-safe. Specifically, The Maximum Permissible Exposure (MPE), which states the highest per-

missible power density [W/cm^2] of $1.4\,\mu\mathrm{m}$ lasers, is about three orders of magnitude larger than our green lasers for our system's typical T^{dot} (*i.e.*, $0.01\,\mathrm{sec}$) [9]. This is because the cornea and lens of the eye strongly absorb wavelengths longer than $1.4\,\mu\mathrm{m}$, preventing them from reaching the retina, which is far more sensitive [10]. Thus, using a $1.4\,\mu\mathrm{m}$ to $1.5\,\mu\mathrm{m}$ lasers with the powers detailed in Sec. 1.1 could be operated in an eye-safe regime.

2. System Calibrations

Our system requires two calibrations. First, the camera matrix and distortion coefficients are computed by capturing a standard black-and-white circle-grid pattern illuminated by a hot Halogen lamp. The lamp raises the black circles' temperature compared to the white background, revealing the pattern in the infrared and allowing the circle detection (*e.g.*, using OpenCV).

Secondly, we calibrate a Homography matrix that maps between the camera's pixel coordinates and the Galvo analog voltages. This is done by projecting and detecting a grid of laser dots on a plane located at our system's typical working distance. The Dual-Axis Galvo consists of two mirrors, which causes the Homography to deviate slightly in planes distant from the calibration plane. However, this distortion had little effect on our experiments since our projection-diffusion reversal network was intentionally trained using noisy predicted point locations in $G(P_{f,m})$.

3. Constructing the G input channel

As described in Section 5.1 of the main paper, to reverse the projection-diffusion of frame $f{+}m$ with respect to frame f, we feed the network with a concatenation of three channels: frame $f{+}m$, heatmap image G, and a constant image with value $0.1m$. In this section, we describe how to construct G.

Image G is a heatmap that 'informs' the network of the predicted spatial locations of the new points $P_{f,m}$. However, as illustrated in Fig. 6 of the main paper, the imaged point locations will deviate from $P_{f,m}$ due to scene motion. Therefore, we construct $G(P_{f,m})$ by assigning projection-order-depended spatial uncertainty to each point.

We construct G by first placing a value of 1 at each projected image point and then convolving each point with a normalized Gaussian whose variance is a monotonically increasing function of the temporal distance $|t_n - t_{f+m}|$:

$$G(P_{f,m}) = \sum_{(\mathbf{x}_n, t_n)_k \in P_{f,m}} \exp\left(-\frac{\mathbf{x}^2}{2\sigma(|t_n - t_{f+m}|)^2}\right) * \delta(\mathbf{x}{-}\mathbf{x}_n). \tag{1}$$

In our training, we set $\sigma(t) = 3 + 2.5t$, where the constants were calibrated empirically. As detailed in Section 4, during training, image G is constructed on the fly for every training example with slightly perturbed locations $P_{f,m}$. Therefore, to speed up the training process, we precompute all the normalized Gaussian kernels in Eq. (1) and efficiently insert them into G using addition.

4. Additional Training Details

This section provides additional technical details about the projection-diffusion reversal network training procedure. As mentioned in Section 5.2 of the main paper, the raw $16\,\mathrm{bit}$ readouts from the thermal camera are scaled to a $[0, 1]$ domain using Eq. (9). We set the constants a and b in Eq. (9) using:

$$\begin{aligned} b &= I^{\max} - I^{\min} + 600 \\ a &= I^{\min} - 200, \end{aligned} \tag{2}$$

where $I^{\max}$ and $I^{\min}$ are the 99.999 and 0.00001 percentile pixel values over all dataset frames. The margin constants in Eq. 2 were added for robustness. In the learning and testing stages, the thermal frames were further scaled to a $[-1, 1]$ range using $I(\mathbf{x}, f) \leftarrow 2(I(\mathbf{x}, f) - 0.5)$.

We used the following data augmentations during training. First, to emulate different ambient temperatures during training, we shift the intensity of each training frame pair:

$$\begin{aligned} I(\mathbf{x}, f) &\leftarrow I(\mathbf{x}, f) + \beta \\ I(\mathbf{x}, f{+}m) &\leftarrow I(\mathbf{x}, f{+}m) + \beta, \end{aligned} \tag{3}$$

where β is uniformly sampled from $[-0.1, 0.1]$. Then, to emulate scene motion, we add a random spatial shift to each point in $P_{f,m}$ when constructing $G(P_{f,m})$. This augmentation forces the network to detect new and undiffuse existing points accurately, even when the heatmap values in $G(P_{f,m})$ do not perfectly align with the newly projected point centers. Finally, we apply the same 128×128 random crop on the three-channel $f + m$ frame and the reference single-channel f frame. The random crop prevents the network from memorizing the pattern point locations. We trained the network for 500 epochs, which lasted about $16\,\mathrm{h}$ on a single GeForce GTX 1080 Ti GPU.

5. Thermal Camera Model Approximation

The Sakuma–Hattori equation of Eq. (1) of the main paper, reproduced below, is a mathematical model for connecting an object's surface temperature $\boldsymbol{T}$ and the raw readout values $S(\boldsymbol{T})$ of a microbolometer thermal camera

$$S(\boldsymbol{T}) = \frac{c_1}{\exp\frac{c_2}{c_3\boldsymbol{T}+c_4} - 1}. \tag{4}$$

This formulation assumes a perfect blackbody and temperatures lower than the melting point of silver [12]. Nevertheless, the form given in Eq. (5) can be used to model real-world camera data by curve fitting the constants $c_1 - c_4$.

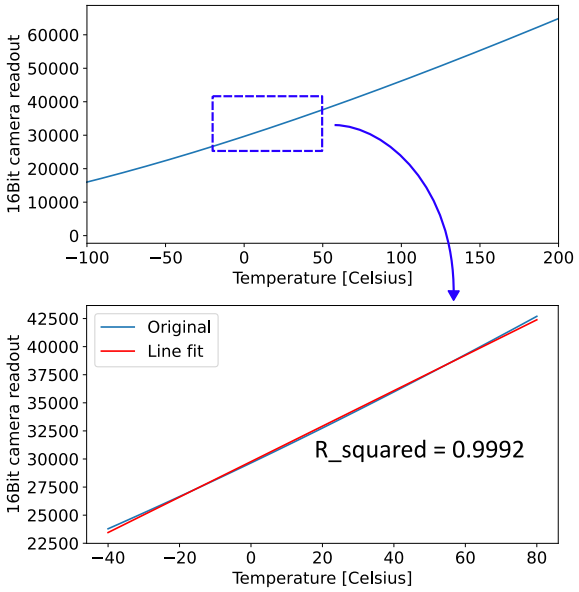

Figure 2. Camera response model. **(Top)** Plot showing a typical thermal camera response [3], modeled by curve fitting with Eq. (5). **(Bottom)** The camera model response is approximately affine in the camera's working range, with a curve fitting yielding a coefficient of determination of 0.9992.

However, as shown in Fig. 2, in the camera's effective temperature range (*i.e.*, $-40°$ to $80°$), the camera model response is approximately affine:

$$S(\boldsymbol{T}) = \frac{c_1}{\exp\frac{c_2}{c_3\boldsymbol{T}+c_4} - 1} \approx c_5\boldsymbol{T} + c_6, \qquad (5)$$

with c_5 and c_6 being some other constants. Importantly, this means that an increase in temperature $\delta\boldsymbol{T}$ yields the same increase in camera readout counts independent of the absolute room temperature. Consequently, our projection-diffusion network must only learn to reverse diffusion based on the local temperature increase (above ambient) and not $\boldsymbol{T}$ itself.

6. Thermal point tracking

In Section 6 of the main paper, we describe how to adapt our method for dynamic vision tasks. Specifically, we explicitly track the projected pattern points between frames to generate point matches for a Structure from Motion solver.

After detection, we track the points using a Lucas-Kanade feature tracker [5]. The projection-diffusion reversal network ensures that the visual appearance of tracked patches is maximally consistent between every two input frames. However, as shown in Fig. 4 of the main paper and Fig. 4 here, the tracked points' SNR decreases with time, which may degrade the point-matching accuracy for long point tracks. Therefore, we track the points as long as their visual appearance exceeds a certain threshold.

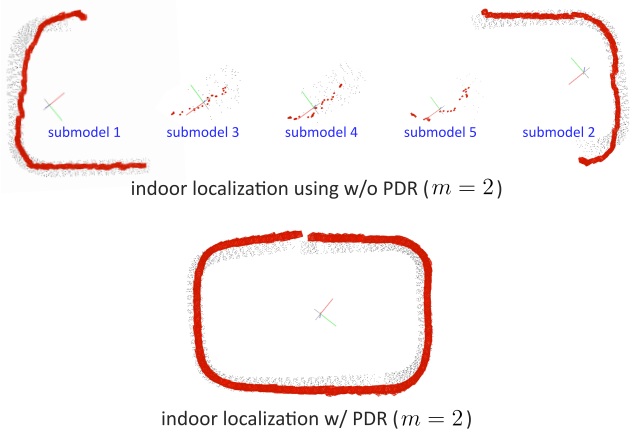

Figure 3. Indoor localization using every other frame. **(Top)** When using every other frame ($m = 2$), the visual difference between frame pairs is large, degrading the SfM reconstruction performance when reconstruction without the projection-diffusion reversal network. **(Bottom)** The PDR network matches the appearance of frame pairs, facilitating a significantly better reconstruction.

Our experiments used a zero mean cross-correlation (ZNCC) score to measure the patches' visual quality [6]. Specifically, for each point, we store a 15×15 pixel patch around the point from the frame at which the point was first detected. Then, for each subsequent frame, we compute a ZNCC score between the initially stored patch and the patch around the currently predicted point location. For each point, tracking is continued as long as the point's ZNCC score is above 0.75.

7. Additional Experimental Results

7.1. Indoor localization using $m = 2$

Fig. 9 of the main paper shows an indoor localization result using our system. The experimental result shown in Fig. 9 was computed using *consecutive* video frames as input to the point tracking algorithm, namely $m = 1$. The floor carpet material on which the patterns were projected exhibited a low thermal diffusivity, yielding a slight change in point appearance between consecutive frames (at 30 Hz). Moreover, the projected pattern points were spread across the entire camera's field of view, creating a relatively sparse pattern in which the newly projected points are easily isolated (even without the PDR's assistance). For these two factors, applying our method without correcting the frames using the projection-diffusion reversal (PDR) network yielded comparable results.

However, as shown in Fig. 3 here, running the reconstruction using *every other frame* ($m = 2$) immediately reveals the PDR network's effectiveness. Specifically, the change in appearance between every other frame was suffi-

ciently large to affect the tracking stage such that COLMAP failed in reconstructing the scene, yielding five separate submodels that could not be merged using COLMAP's `model_merger` module (see Fig. 3(Top)). Conversely, as seen in Fig. 3(Bottom), applying the PDR network for $m = 2$ produced a similarly accurate result to the original result in Fig. 9 of the main paper.

The comparison in Fig. 3 is another example of the PDR network's effectiveness for tracking thermal patterns when prior vision frameworks are used. While in this particular experiment, using $m = 2$ was unnecessary, this capability is beneficial for several reasons, even for surfaces like the carpet in Fig. 9 of the main paper. For instance, U.S. government export control requires selling the camera used in this paper to various countries with reduced frame rates (*i.e.*, 9 Hz), which would necessitate a larger diffusion reversal [4] (equivalent to $m = 3$ compared to 30 Hz mode). Moreover, real-world applications may suffer from occasional corrupt frames due to camera shakes, momentary occlusions, or flashing of external heat sources that would require a robust system to match between non-consecutive frames.

7.2. Laser absorption vs. object albedo

In Fig 4, we tested the laser's absorption on various albedos. In Fig 4(Left), we projected dots having the same duration on the neutral patches of a standard color chart. The plot shows the imaged maximum temperature rise in the percentage of the black albedo patch (*i.e.*, bottom right patch). As expected from Eq. (2) of the main paper, the temperature rise is proportional to the albedo. Fig 4(Right) shows an imaged pattern projected on a white office wall using the 1 W laser with a $T^{\text{dot}} = 16.6$ ms (*i.e.*, $K = 2$), while Fig 4(Bottom) shows the SNR plot of a single wall point. Fig 4 shows that the projected pattern is visible and trackable even on a white wall.

7.3. Poorly behaving materials

As mentioned in Section 9 of the main paper, our method relies on remotely heating local surface patches on scene objects. The laser heating process depends on various material properties, such as the material's absorption of the laser's wavelength (*i.e.*, albedo), thermal conductivity and diffusivity, and emissivity in the camera's infrared range. Fig. 5 shows various examples of materials where the properties above for these materials yield poor performance, namely, low laser heating and fast point diffusion.

For example, objects made of metal (*e.g.*, the "REI cup", "water flask" and "drink can" in Fig. 5), even if painted black, will behave poorly due to the metal's high thermal conductivity, causing the heated points to diffuse rapidly and last only a few frames (and even less than one frame as seen in Fig. 5). Moreover, objects having low emissivity

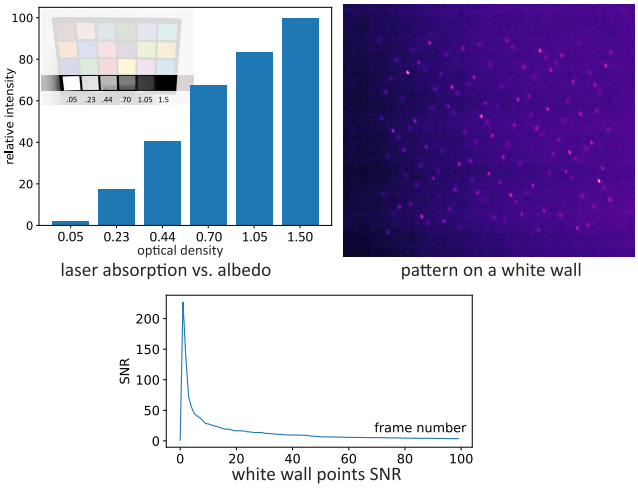

Figure 4. Laser absorption vs. object albedo. **(Left)** Plot showing the relative measured signal intensity for patches of varying albedo. The intensity is relative to the bottom-right black patch. **(Right)** A single thermal frame of a pattern projected on a white office wall with $K = 2$. **(Bottom)** SNR plot for a single point wall pattern point in (Right).

(*e.g.*, the "RI mug" and 'drink can' in Fig. 5) will mostly reflect and not absorb (and emit back) the laser radiation. This is evident when observing the "drink can" thermal frames. The can is made out of aluminum, which has a notable low emissivity [8], manifesting in the strong specular reflections of the surrounding objects, as seen in Fig. 5. Notably, for the can, the projected laser dots are entirely imperceptible.

References

[1] Oem cameras - flir boson series. https://www.oemcameras.com/thermal-imaging-cameras/thermal-imaging-cores/flir-boson-series.htm/22640A024.htm. 1

[2] Oxlasers green laser module. https://tinyurl.com/32cdpzzd. 1

[3] Teledyne flir. https://flir.custhelp.com/app/answers/detail/a_id/3321/~/the-measurement-formula. 3

[4] Teledyne flir export information. https://flir.custhelp.com/app/answers/detail/a_id/3221/~/oem-camera-modules-export-information. 4

[5] Jean-Yves Bouguet et al. Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel corporation*, 5(1-10):4, 2001. 3

[6] Kai Briechle and Uwe D Hanebeck. Template matching using fast normalized cross correlation. In *Optical Pattern Recognition XII*, volume 4387, pages 95–102. SPIE, 2001. 3

[7] Dorian Chan, Mark Sheinin, and Matthew O'Toole. Spincam: High-speed imaging via a rotating point-spread func-

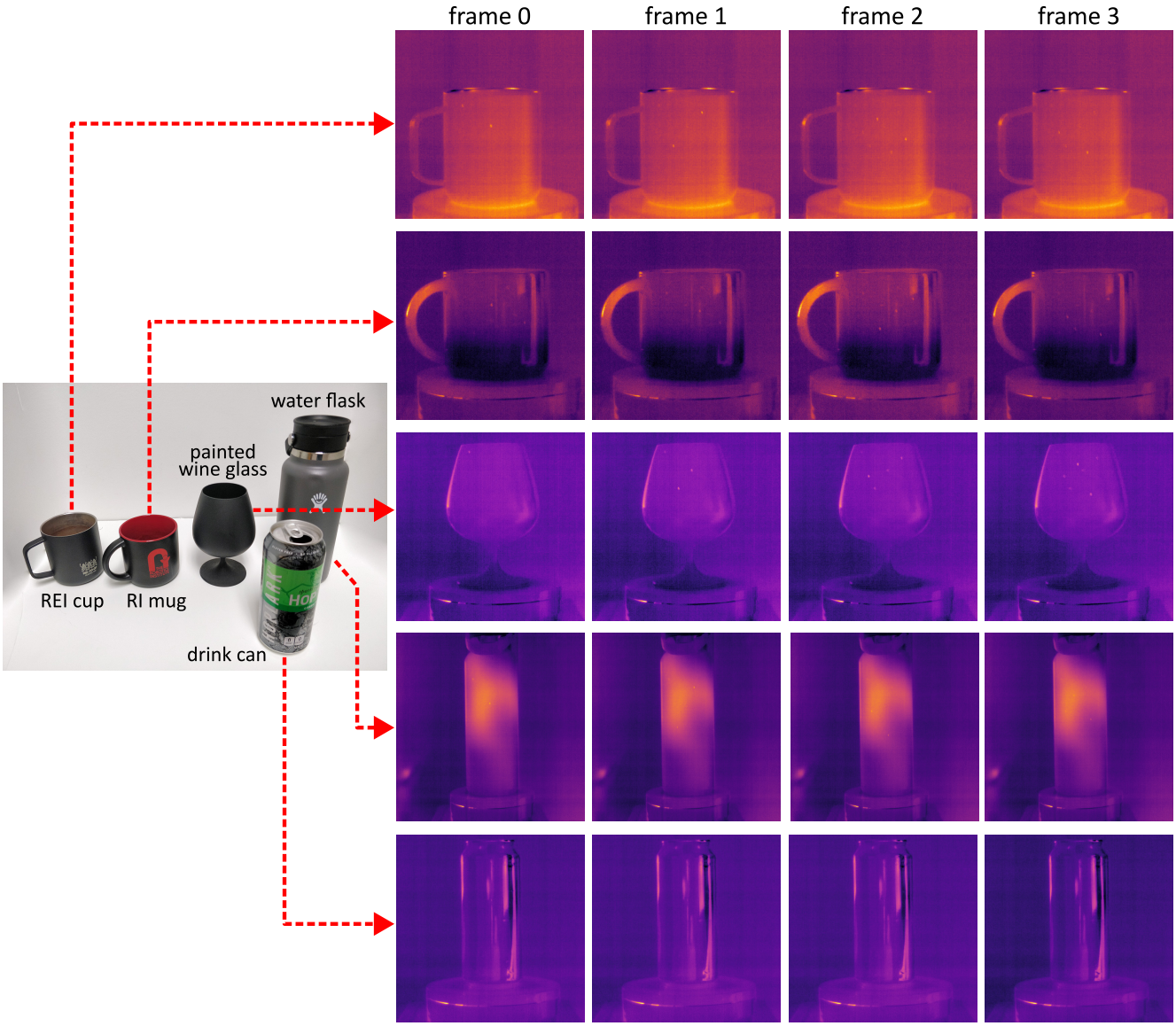

Figure 5. Various examples of objects with materials yielding poor thermal pattern trackability performance.

tion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10789–10799, 2023. 1

[8] Wikipedia contributors. Emissivity. https://en.wikipedia.org/wiki/Emissivity. 4

[9] Wikipedia contributors. Laser safety, section: Maximum permissible exposure. https://en.wikipedia.org/wiki/Laser_safety. 2

[10] RP Photonics. Eye-safe lasers. https://tinyurl.com/4ftuhrzz. 2

[11] Manikandasriram Srinivasan Ramanagopal, Zixu Zhang, Ram Vasudevan, and Matthew Johnson-Roberson. Pixel-wise motion deblurring of thermal videos. *arXiv preprint arXiv:2006.04973*, 2020. 1

[12] Fumihiro Sakuma and Susumu Hattori. Establishing a practical temperature standard by using a narrow-band radiation thermometer with a silicon detector. *Metrology Institute Report*, 32(2):p91–97, 1983. 2