HOT-POT: Optimal Transport for Sparse Stereo Matching

Antonin Clerc[1,2*†], Michael Quellmalz[1*†], Moritz Piening[1*†], Philipp Flotho[3,4], Gregor Kornhardt[1], Gabriele Steidl[1]

[1]Institute of Mathematics, Technische Universität Berlin, Germany.
[2]Univ. Bordeaux, CNRS, Bordeaux INP, IMB, UMR 5251, F-33400 Talence, France.
[3]Chair for Clinical Bioinformatics, Saarland Informatics Campus, Saarland University, Germany.
[4]Optical Neuroimaging Unit, Okinawa Institute of Science and Technology, Japan.

*Corresponding author(s). E-mail(s): antonin.clerc@math.u-bordeaux.fr; quellmalz@math.tu-berlin.de; piening@math.tu-berlin.de;
Contributing authors: Philipp.Flotho@uni-saarland.de; kornhardt@math.tu-berlin.de; steidl@math.tu-berlin.de;
†These authors contributed equally to this work.

Abstract

Stereo vision between images faces a range of challenges, including occlusions, motion, and camera distortions, across applications in autonomous driving, robotics, and face analysis. Due to parameter sensitivity, further complications arise for stereo matching with sparse features, such as facial landmarks. To overcome this ill-posedness and enable unsupervised sparse matching, we consider line constraints of the camera geometry from an optimal transport (OT) viewpoint. Formulating camera-projected points as (half)lines, we propose the use of the classical epipolar distance as well as a 3D ray distance to quantify matching quality. Employing these distances as a cost function of a (partial) OT problem, we arrive at efficiently solvable assignment problems. Moreover, we extend our approach to unsupervised object matching by formulating it as a hierarchical OT problem. The resulting algorithms allow for efficient feature and object matching, as demonstrated in our numerical experiments. Here, we focus on applications in facial analysis, where we aim to match distinct landmarking conventions.

Keywords: Optimal Transport, Hierarchical Optimal Transport, Stereo Matching, Stereo Vision

1 Introduction

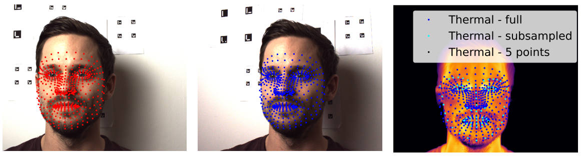
Identification of features across views and inference of depth information via **stereo matching** is a core technique in computer vision, allowing for 3D reconstructions from multiple 2D views [42]. It enables obstacle detection in robotics and autonomous driving, surface defect detection in industrial applications, and head reconstruction in facial analysis [34, 42, 54, 58]. However, practical algorithms need to overcome a variety of real-world challenges.

Even for single-modality camera systems, the identification of view-invariant features is generally hindered by radiometrically distorted pixel brightness, depth changes near object boundaries, and partial occlusions. These issues become more pronounced for cross-modal systems like RGB-thermal. In particular, traditional stereo matching methods relying on the comparison of pixel values [12, 16, 24] become inapplicable under these conditions. Therefore, suitable cross-spectral approaches are mostly based on pretrained neural networks [43, 32, 52, 71] and local feature descriptors [40, 72]. Notably, most modern feature extractors are dense [26, 38], producing one feature per pixel. In contrast, we focus on **sparse features** located at image keypoints [59].

Beyond the detection of feature descriptors, their cross-view matching is inherently ill-posed since it is highly sensitive to detector noise and occlusions [45, 52], especially if only the location and no additional information, such as color, is available. Moreover, this instability increases dramatically for sparse features as considered in this work. In this setting, key challenges arise from significant differences between descriptors across views in terms of their number and their locations. Such limitations necessitate the modification of nearest-neighbor matching to ensure robustness [59].

Optimal transport (OT) provides a robust relaxation of nearest-neighbor matching [49, 67], relaxing the optimal assignment problem to a linear program over probabilistic assignments. Due to the availability of efficient solvers via Sinkhorn's method [15, 20, 33] or slicing [7, 35, 46, 56, 57] as well as extensions to partial assignments [13, 14, 61], the OT framework has become a popular tool for comparing and matching point clouds [37, 39, 55, 62]. This has further led to applications in stereo vision with dense features by matching along image rows using OT [25] and by using an OT-based module in the stereo matching model H-Net [29], which allows the network to focus on feature mismatches regarding the so-called epipolar constraints of the camera geometry. Extending this approach, we derive distances between feature keypoints from these epipolar constraints to perform sparse stereo matching using OT assignments. Beyond feature matching via **classical OT**, we integrate the resulting cost functions into **hierarchical OT** [1, 18, 47, 60] to enable object matching based on sparse keypoints.

As a particular application, the main motivation for this study is a **cross-modal stereo vision** setting similar to [21]. We are given facial image pairs captured simultaneously by a conventional RGB camera and a thermal (long-wave infrared) camera, whose intrinsic parameters and locations are known. The main goal is the matching of facial features obtained by pretrained modality-dependent feature trackers or **landmarkers** [10, 31, 66, 68, 69]. These landmarkers discretize the underlying 3D facial geometry and may yield distinct sets of points for different modalities [19, 23], see Figure 1. Therefore, the two sets of 2D points are subject to modality-specific noise and may differ in cardinality. Furthermore, for two stereo images containing faces of many persons, we want to identify reliably which face in the RGB image belongs to which face in the thermal image.



1st RGB 2nd RGB Thermal

Fig. 1: RGB (left, middle) and thermal (right) images with facial landmarks. Landmarks conventions vary in terms of size and locations, as illustrated by the thermal image. Images were originally reported in [22].

We model our setup via an unknown 3D point cloud that is projected onto both camera planes, yielding two distinct 2D point clouds, which are considered as our measurements. Our practical goal is then i) to perform an accurate point-to-point matching across camera planes and ii) to establish correspondences between entire objects, each containing multiple points. In practice, these 2D point clouds are obtained via given feature tracking algorithms, such as a landmarker. Within the setup of facial landmarking, we pursue the goal of cross-modal landmark-to-landmark and face-to-face matchings.

Our main contributions are as follows:

- We formulate the stereo matching problem as an instance of an OT problem [49, 67] with cost functions tailored to the geometric setting.
- We propose a novel cost function, called 3D ray distance, based on the distance between

3D half-lines (rays) originating from the camera centers. Numerical experiments show its advantage over the classical epipolar distance [27, 54].

- To handle matching between entire objects (e.g., faces) represented as unordered sets of points, we introduce a **hierarchical optimal transport** (HOT) formulation. Numerical studies demonstrate that HOT yields stable correspondences.

Our method addresses several of the classical challenges in stereo vision. The proposed cost function enhances the robustness of point-level matching by reducing sensitivity to noise and ambiguity, and by limiting the set of admissible matches. This proves to be particularly effective for occlusions and cross-modality scenarios. The use of OT offers an efficient formulation for the matching problem. Finally, the HOT framework extends our algorithm from point-to-point to object-to-object matching.

The paper is organized as follows. Section 2 covers the epipolar geometry and introduces our new matching cost. Section 3 introduces basic concepts of OT required to understand the framework, and outlines the specific OT problems we consider. Section 4 focuses on algorithmic considerations, including numerical implementation, error computation, and evaluation metrics to assess the quality of the matchings. Section 5 presents numerical results on various simulated and real-world datasets both for OT and HOT formulations. Conclusions are drawn in Section 6.

2 Distances of Projected Points

In this section, we propose two different "distances" between projected points in the camera planes which aim to preserve their true, unknown distances in 3D, namely

- the 3D ray distance and
- the epipolar distance.

For a detailed treatment of the underlying epipolar geometry and camera models, we refer to [28, 54, 63]. Considering two cameras observing the same 3D point $w \in \mathbb{R}^3$, we can express the points measured by the cameras in homogeneous coordinates with the third component fixed to one, i.e., as elements of the projective space

$$\mathbb{P}^2 := \{(x_1, x_2, 1) \mid (x_1, x_2) \in \mathbb{R}^2\}.$$

We choose the coordinate system such that the left camera is centered at $(0,0,0)^\top$ and is imaging along the positive third coordinate. By K_l, $K_r \in \mathbb{R}^{3\times 3}$ we denote the **intrinsic matrix** of the left and right camera, respectively. These are upper triangular matrices with positive diagonal entries, which encode internal characteristics such as focal length and principal point of the camera. The **extrinsic parameters** describe the relative orientation and position between the two cameras, modeled by a rotation matrix $R \in \mathrm{SO}(3)$ and a translation vector $t \in \mathbb{R}^3 \setminus \{0\}$. Then the projections $x, y \in \mathbb{P}^2$ of a point $w \in \mathbb{R}^3$ onto the left and right camera plane are given by

$$\lambda_l x = K_l w \quad \text{and} \quad \lambda_r y = K_r(Rw + t), \tag{1}$$

respectively, where $x, y \in \mathbb{P}^2$ and $\lambda_l, \lambda_r > 0$ denote the third component of $K_l w$ and $K_r w$, respectively. We assume that w is located in front of both cameras, meaning that

$$\langle w, e_3 \rangle > 0 \quad \text{and} \quad \langle Rw + t, e_3 \rangle > 0. \tag{2}$$

The configuration is shown in Figure 2. Let

$$\mathcal{W} := \{w \in \mathbb{R}^3 : \langle w, e_3 \rangle > 0,\ \langle Rw + t, e_3 \rangle > 0\}.$$

Remark 2.1. *In a more general model, where each camera has a rotation $R_l, R_r \in \mathrm{SO}(3)$ and translation $t_r, t_l \in \mathbb{R}^3 \setminus \{0\}$, the projections are given by*

$$\begin{aligned} \lambda_l x &= K_l(R_l \tilde{w} + t_l), \\ \lambda_r y &= K_r(R_r \tilde{w} + t_r). \end{aligned} \tag{3}$$

Then the substitution $\tilde{w} = R_l^\top (w - t_l)$, and setting $R = R_r R_l^\top$ and $t = t_r - R_r t_l$ yields (1).

2.1 3D Ray Distance

Setting

$$\tilde{x} := K_l^{-1} x \quad \text{and} \quad \tilde{y} := K_l^{-1} y,$$

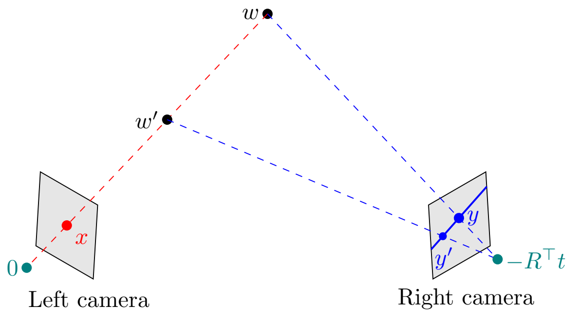


Fig. 2: 3D point w observed by two cameras, specifically at x in the left and y in the right camera. The epipolar line (solid blue) in the right camera corresponding to x consists of all points y' that originate from 3D points w' that are projected to x in the left camera. In teal are the focal points 0 and $-R^\top t$.

we obtain by (1) that

$$w = \lambda_l \tilde{x} \quad \text{and} \quad w = \lambda_r R^\top \tilde{y} - R^\top t. \tag{4}$$

The right-hand side of each of the two equations represents a line in $\mathbb{R}^3$ of the form

$$\begin{aligned} \mathcal{L}_x &= \{\lambda_x r_x + s_x : \lambda_x \in \mathbb{R}\}, \\ \mathcal{L}_y &= \{\lambda_y r_y + s_y : \lambda_y \in \mathbb{R}\}, \end{aligned} \tag{5}$$

where $r_x = \tilde{x}$, $s_x = 0$, $r_y = R^\top \tilde{y}$, and $s_y = -R^\top t$. The minimal distance between these lines, $\mathrm{d}(\mathcal{L}_x, \mathcal{L}_y)$, is given by

$$\begin{cases} \frac{|\langle r_x \times r_y, s_y - s_x \rangle|}{\|r_x \times r_y\|} = \frac{|\langle \tilde{x} \times R^\top \tilde{y}, R^\top t \rangle|}{\|\tilde{x} \times R^\top \tilde{y}\|} & \text{if } r_x \times r_y \neq 0, \\ \frac{\|r_x \times (s_y - s_x)\|}{\|r_x\|} = \frac{\|\tilde{x} \times R^\top t\|}{\|\tilde{x}\|} & \text{otherwise} \end{cases}$$

and could serve as a possible distance between x and y. Note that $r_x \times r_y = 0$ if and only if $\mathcal{L}_x \parallel \mathcal{L}_y$. However, this distance is computed on the entire space $\mathbb{R}^3$, but we are only interested in intersections or minimal distances for points that are visible to both cameras. Indeed, the lines (5) may intersect or attain their minimal distance behind the cameras. In such cases, although the distance between the lines is small, the corresponding point is not observable by the cameras. Therefore, such matches should be avoided.

To this end, consider the case $r_x \times r_y \neq 0$. Then there exists a unique shortest line segment connecting $\mathcal{L}_x$ and $\mathcal{L}_y$. It intersects the respective lines in

$$\begin{aligned} b_x &= s_x + \frac{\langle s_x - s_y, n_x \rangle}{\langle r_x, n_x \rangle} r_x, \\ b_y &= s_y - \frac{\langle s_x - s_y, n_y \rangle}{\langle r_y, n_y \rangle} r_y, \end{aligned} \tag{6}$$

where $n_x := r_x \times (r_x \times r_y)$ and $n_y := r_y \times (r_x \times r_y)$. Both points b_x, b_y should fulfill (2). If b_x or b_y lies behind one of the cameras, the ray distance is usually smallest between the focal points of the two cameras. Hence, we use the distance between the focal points, namely $\|R^\top t - 0\| = \|t\|$. Thus, we define **ray distance** $d^{\text{ray}} : \mathbb{P}^2 \to \mathbb{R}_{\geq 0}$ by

$$d^{\text{ray}}(x, y) := \begin{cases} \frac{|\langle \tilde{x} \times R^\top \tilde{y}, R^\top t \rangle|}{\|\tilde{x} \times R^\top \tilde{y}\|} & \text{if } r_x \times r_y \neq 0, \\ & \text{and } b_x, b_y \in \mathcal{W}, \\ \frac{\|\tilde{x} \times R^\top t\|}{\|\tilde{x}\|} & \text{if } r_x \times r_y = 0, \\ \|t\| & \text{otherwise.} \end{cases} \tag{7}$$

The so-defined ray distance is not a metric, because, in general (depending on the camera parameters), d^{ray} is not symmetric and $d^{\text{ray}}(x, x)$ may be non-zero. However, we have the following properties.

Proposition 2.2. *If $w \in \mathcal{W}$ and $x, y \in \mathbb{P}^2$ are given by (1), then $d^{\text{ray}}(x, y) = 0$. Conversely, if $x, y \in \mathbb{P}^2$ and $d^{\text{ray}}(x, y) = 0$, there exists $w \in \mathbb{R}^3$ such that (1) holds for some $\lambda_l, \lambda_r \in \mathbb{R}$. If additionally $r_x \times r_y \neq 0$, then w is uniquely determined.*

Proof Let w satisfy (2) and x, y be the projections of w via (1). By construction, w is on both lines $\mathcal{L}_x$ and $\mathcal{L}_y$. If the lines are not identical, i.e., if $r_x \times r_y \neq 0$, we have $w = b_x = b_y$. By (2), we are in the first case of the definition of d^{ray} and hence we have $d^{\text{ray}}(x, y) = 0$. Otherwise, if the lines are identical, i.e., if $r_x \times r_y = 0$, their distance is zero and so $d^{\text{ray}}(x, y) = 0$.

Conversely, let $d^{\text{ray}}(x, y) = 0$. We note that $\|t\|$ does not vanish by assumption. If $0 = r_1 \times r_2 = \tilde{x} \times R^\top \tilde{y}$, we have $\tilde{x} \times R^\top t = 0$. Hence, $\tilde{x}$, $R^\top \tilde{y}$, and $R^\top t$ are all located one line through the origin and neither does vanish. Hence, (4), which is equivalent to (1), is fulfilled for $w = \alpha \tilde{x}$ with any $\alpha \neq 0$ and some $\lambda_l, \lambda_r \in \mathbb{R}$. Otherwise, the lines intersect in one point $b_x = b_y$, and hence (4) is fulfilled with $w = b_x = b_y$. In this case, w satisfies (2). □

Remark 2.3 (Depth-regularized Ray Distance). *While Proposition 2.2 justifies the use of the ray distance theoretically, the computation of the ray* *distance is practically sensitive to perturbations of the camera parameters. To improve stability, a common remedy is the use of regularization to include prior knowledge [5]. In most practical scenarios, we have prior information about the possible depth of the objects, e.g., if the camera is located in a room, the objects cannot be farther away than the walls. Motivated by this, we consider lower and upper soft thresholds $\gamma_1 < \gamma_2$ on the depth and some regularization parameter $\beta \geq 0$. Then, we modify our distance (7) to introduce the* ***depth-regularized ray distance*** *$d^{\mathrm{reg}}_{\beta,\gamma}(x,y)$ defined as*

$$d^{\mathrm{ray}}(x,y) + \beta \begin{cases} (b-\gamma_1)^2 & \text{if } b < \gamma_1, \\ 0 & \text{if } b \in [\gamma_1, \gamma_2], \\ (b-\gamma_2)^2 & \text{if } b > \gamma_2, \end{cases} \tag{8}$$

where $b := \frac{1}{2}\langle b_x + b_y, e_3\rangle$ is the third coordinate of the midpoint of b_x, b_y given in (6).

Remark 2.4 (Invariance to Rotations)**.** *The ray distance is invariant to rotations of the camera with the same focal point. More specifically, taking a different right camera with parameters $R' \in SO(3)$ and $t' \in \mathbb{R}^3$ and the same focal point $R^\top t = (R')^\top t'$. Then the ray distance $d^{\mathrm{ray}}(x,y)$ coincides with the ray distance between x and y' with respect to the other camera, where y' is the normalized projection of $w \in \mathbb{R}^3$ fulfilling $w = \lambda'_r (R')^\top \tilde{y}' - (R')^\top t'$, cf. (4).*

2.2 Epipolar Distance

The second distance arises from epipolar (half)lines, which are often used in the literature, see, e.g. [17, 30, 53]. Substituting $w = \lambda_l \tilde{x}$ into the second equation in (4) gives

$$\lambda_r \tilde{y} = \lambda_l R\tilde{x} + t.$$

Taking the cross product of both sides with the translation vector $t = (t_1, t_2, t_3)^\top$ results in

$$t \times (\lambda_r \tilde{y}) = t \times (\lambda_l R\tilde{x}), \tag{9}$$

which can be rewritten as

$$\lambda_r T\tilde{y} = \lambda_l TR\tilde{x}, \qquad T := \begin{pmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{pmatrix}.$$

Since the cross product is perpendicular to its generating vectors, we obtain by taking the inner product with $\tilde{y}$ the **epipolar constraint**

$$y^\top F x = 0, \qquad F := K_r^{-\top} TRK_l^{-1},$$

where F is called **fundamental matrix**. The **epipolar line with respect to** $x \in \mathbb{P}^2$ in the right camera plane is given by

$$\{(s_1, s_2)^\top : \langle Fx, s\rangle = 0\}, \; s := (s_1, s_2, 1)^\top. \tag{10}$$

Geometrically, the epipolar line is the projection of the line $\mathcal{L}_x$ in (5) to the right camera plane, see Figure 2. Then the distance of a point $y \in \mathbb{P}^2$ in the right image to the epipolar line (10) of $x \in \mathbb{P}^2$ is given by

$$d^{\mathrm{epi}}_r(x,y) := \begin{cases} \dfrac{|\langle Fx, y\rangle|}{\|P_2 Fx\|} & \text{if} \quad P_2Fx \neq 0, \\ |(Fx)_3| & \text{if} \quad P_2Fx = 0, \end{cases}$$

where $P_2(x_1,x_2,x_3) := (x_1,x_2)$ and $(Fx)_3$ denotes the third component of the vector. The second case is motivated by the fact that if $P_2Fx = 0$, we have $Fx = (0,0,(Fx)_3)^\top$ and hence $\langle Fx, y\rangle = (Fx)_3$ for all $y \in \mathbb{P}^2$. Then the epipolar line (10) degenerates to either the plane $\mathbb{P}^2$ if $(Fx)_3 = 0$, or the empty set if $(Fx)_3 \neq 0$. By the next remark, the case $P_2Fx = 0$ cannot appear for points in front of the camera and if the cameras do not see each other.

Remark 2.5. *Assume that $P_2Fx = 0$. This implies $Fx = \alpha e_3$ with $\alpha = (Fx)_3$ and since K_r is upper triangular further*

$$\begin{aligned} Fx = K_r^{-\top} TRK_l^{-1} x &= \alpha e_3, \\ TRK_l^{-1} x &= \alpha (K_r)_{3,3} e_3, \\ TRw = t \times Rw &= \lambda_l \alpha (K_r)_{3,3} e_3. \end{aligned}$$

If $\alpha \neq 0$, this means that

$$\langle Rw, e_3\rangle = \langle t, e_3\rangle = \langle R^\top t, R^\top e_3\rangle = 0,$$

and consequently $\langle Rw + t, e_3\rangle = 0$ which contradicts (2). Geometrically, all such w are in the plane with normal direction $R^\top e_3$ through the focal point $-R^\top t$ of the second camera. If $\alpha = 0$, then w is a multiple of $-R^\top t$ and lies therefore on the line between the two focal points, see

Figure 2. Then the epipolar line degenerates to a single point. In practice, this case does usually not occur as it would mean that the second camera is visible in the first camera image.

Similarly, we can consider the distance of a point $x \in \mathbb{P}^2$ in the left image to the epipolar line of $y \in \mathbb{P}^2$ to obtain the **epipolar distance** in the left image

$$d_l^{\text{epi}}(x,y) \coloneqq \begin{cases} \dfrac{|\langle x, F^\top y\rangle|}{\|P_2 F^\top y\|} & \text{if} \quad P_2 F^\top y \neq 0, \\ |(F^\top y)_3| & \text{if} \quad P_2 F^\top y = 0. \end{cases}$$

Averaging over the epipolar distances in the left and right image, we get the final **epipolar distance**

$$d^{\text{epi}}(x,y) \coloneqq \frac{d_l^{\text{epi}}(x,y) + d_r^{\text{epi}}(x,y)}{2}, \tag{11}$$

in particular

$$d^{\text{epi}}(x,y) = \frac{|\langle Fx, y\rangle|}{2}\left(\frac{1}{\|P_2 Fx\|} + \frac{1}{\|P_2 F^\top y\|}\right) \quad \text{if} \quad P_2 F x \neq 0 \text{ and } P_2 F^\top y \neq 0.$$

Remark 2.6 (Epipolar rays)**.** *Instead of the line (5), we could project the ray located in front of the camera to the image plane, leading to an epipolar ray, cf. [17, 30, 53]. The condition (2), which means a point is in front of both cameras, is equivalent to* $\lambda_1, \lambda_2 > 0$ *in (1). By (9), we have*

$$\tfrac{\lambda_2}{\lambda_1} t \times \tilde{y} = t \times R\tilde{x},$$

which holds for $\lambda_1\lambda_2 > 0$ *if and only if*

$$\langle t \times \tilde{y}, t \times R\tilde{x}\rangle > 0 \tag{12}$$

or $t \times \tilde{y} = t \times R\tilde{x} = 0$. *We define for fixed* $x \in \mathbb{P}^2$ *the epipolar ray* H_x *in the right image by*

$$\{y \in \mathbb{P}^2 : \langle y, Fx\rangle = 0,\ \langle t \times K_r^{-1} y, t \times R K_l^{-1} x\rangle > 0\}.$$

We show that in most practical scenarios, where both cameras depict a similar region of the 3D space, the epipolar rays and epipolar lines coincide inside the images. More specifically, we assume the camera image is a square

$$I_a \coloneqq [-a,a]^2 \times \{1\} \subset \mathbb{P}^2, \qquad a > 0.$$

The epipole $y_{\text{e}} \in \mathbb{P}^2$ *is the projection of the focal point* $w = 0$ *of the left camera to the right image given by*

$$\lambda_l x_{\text{e}} = -K_l R^\top t.$$

We assume that y_{e} *is not visible in the right camera, i.e.,* $y_{\text{e}} \notin I_a$, *and that* $w \in \mathbb{R}^3$ *is visible by both cameras with the projections* $x \in \mathbb{P}^2$ *and* $y \in I_a$. *Then*

$$H_x \cap I_a = \{y \in I_a : \langle y, Fx\rangle = 0\}.$$

This can be seen as follows. By definition, the left-hand side is a subset of the right. We show that (12) is fulfilled for all $y \in I_a$ *with* $\langle Fx, y\rangle = 0$. *The function* $y \mapsto \langle t \times \tilde{y}, t \times R\tilde{x}\rangle$ *is affine-linear, hence its zero set is affine-linear in* $\mathbb{P}^2$ *and it contains the epipole* y_{e}. *Therefore,* H_x *is either a ray starting at the epipole* y_{e} *or empty if (12) vanishes on the whole epipolar line. Hence, the intersection* $H_x \cap I_a$ *is either* $I_{a,b} \cap \{y \in \mathbb{P}^2 : \langle y, Fx\rangle = 0\}$ *or empty. The latter cannot hold, as the set contains the projection of* w.

3 Optimal Transport

We recall basic notions of discrete OT and its partial version as well as hierarchical OT, cf. [49]. While we formulate the OT problem for discrete measures on the projective plane $\mathbb{P}^2$, the theory in this section also applies to continuous probability measures on general manifolds.

3.1 Point Matching via Optimal Transport

Given two sets

$$X = \{x^1, \ldots, x^N\} \in (\mathbb{P}^2)^N,$$
$$Y = \{y^1, \ldots, y^M\} \in (\mathbb{P}^2)^M,$$

we are interested in the optimal transport costs between the empirical measures

$$\mu = \frac{1}{N}\sum_{i=1}^N \delta_{x^i} \quad \text{and} \quad \nu = \frac{1}{M}\sum_{i=1}^M \delta_{y^i},$$

where δ_x is the point measure at x, defined by the discrete **OT distance**

$$\begin{aligned} &\mathrm{OT}_c(X,Y) \coloneqq \min_{\Pi\geq 0} \langle C, \Pi\rangle \\ &\text{subject to} \quad \Pi \mathbf{1}_M = \frac{1}{N}\mathbf{1}_N,\ \Pi^\top \mathbf{1}_N = \frac{1}{M}\mathbf{1}_M, \end{aligned} \tag{13}$$

where $c\colon \mathbb{P}^2 \times \mathbb{P}^2 \to [0,\infty[$ is a cost or "distance function" on $\mathbb{P}^2$,

$$C \coloneqq \big(c(x^i, y^j)\big)_{i,j=1}^{N,M} \quad \text{and} \quad \Pi \coloneqq (\pi_{i,j})_{i,j=1}^{N,M}.$$

The matrix $\Pi \in \mathbb{R}_{\geq 0}^{N\times M}$ is called a transport plan. In our applications, we will deal with the "distances" d^{ray}, $d^{\mathrm{reg}}_{\beta,\gamma}$ and d^{epi} from the previous section. If $N = M$, there exists an optimal solution Π such that $N\Pi$ is a permutation matrix associated with a permutation $\sigma \in \mathrm{Perm}(N)$ and $\mathrm{OT}_c(X,Y) = \sum_{i=1}^N c(x^i, y^{\sigma(i)})$, see [49, Prop 2.1].

However, in a practical stereo matching application, object occlusions or modality-dependent feature trackers might violate the assumption of the balanced setup $N = M$. Clearly, for $N \neq M$, a transport plan cannot be realized by a permutation matrix. Therefore, we consider the more general **partial optimal transport** (POT) problem [11, 13], which explicitly restricts the total transported mass and is defined as

$$\begin{aligned} &\mathrm{POT}_c(X,Y) \coloneqq \min_{\Pi\geq 0} \langle C, \Pi\rangle \\ &\text{subject to} \quad \mathbf{1}_N^\top \Pi\, \mathbf{1}_M = m, \\ &\Pi \mathbf{1}_M \leq \frac{1}{N}\mathbf{1}_N, \quad \Pi^\top \mathbf{1}_N \leq \frac{1}{M}\mathbf{1}_M, \end{aligned} \tag{14}$$

for a mass constraint $m \in [0,1]$. While it simplifies to (13) for $N = M$ and $m = 1$, it allows excess mass to be discarded for $N \neq M$. Consequently, POT is especially well-suited for computing a partial correspondence between point clouds resulting from incompatible landmarkers. If $m = \min\{N,M\}/\max\{N,M\}$, there exists an optimal solution $\Pi/m \in \{0,1\}^{N\times M}$ such that Π is associated with a 'partial' permutation between the smaller set and an equal-sized subset of the larger set, see [3, Thm. 4.1].

Beyond partial matching, one might alternatively replace POT with the closely related and more general **unbalanced optimal transport** formulation [4, 61], where one regularizes the transport problem with respect to a fixed probability divergence, see [64, Appendix A] for examples. However, the latter has two regularization parameters that need to be chosen appropriately.

3.2 Object Matching via Hierarchical Optimal Transport

Hierarchical optimal transport (HOT) refines the constraints of the original OT problem by introducing additional labels. In the following, let $[N] \coloneqq \{1,\dots,N\}$. Now the goal is to match full objects each consisting of many points. Given N source objects X_i and M target objects Y_j, let

$$\begin{aligned} &\boldsymbol{X} = \{X_1,\dots,X_N\} \quad \text{and} \quad \boldsymbol{Y} = \{Y_1,\dots,Y_M\} \\ &\text{with} \quad X_i = \{x_i^1,\dots,x_i^{N_i}\} \in (\mathbb{P}^2)^{N_i},\ i\in[N], \\ &\qquad\quad Y_j = \{y_j^1,\dots,y_j^{M_j}\} \in (\mathbb{P}^2)^{M_j},\ j\in[M]. \end{aligned}$$

We use a two-step hierarchical strategy, inspired by hierarchical Wasserstein distance formulations [1, 18, 70]. In the case $N = M$ and a Euclidean cost function, our hierarchical OT formulation computes a discretized version of the so-called Wasserstein over Wasserstein [6, 51] or the mixture Wasserstein distance [18, 50].

Step 1 (Local pointwise matching between objects): For each pair (X_i, Y_j), $i \in [N]$, $j \in [M]$, we solve the POT problem (14) to obtain

$$c^{\mathrm{obj}}(X_i, Y_j) \coloneqq \mathrm{POT}_c(X_i, Y_j). \tag{15}$$

Step 2 (Global matching): Once the object-to-object cost c^{obj} has been computed, we solve a second POT problem at the object level:

$$\mathrm{POT}_{c^{\mathrm{obj}}}(\boldsymbol{X}, \boldsymbol{Y}). \tag{16}$$

The hierarchical matching procedure is summarized in Algorithm 1. In the case of a balanced matching using (13), we refer to this procedure as **HOT**. If we employ the POT formulation (14), we use the name **HOT-POT**.

Recovering a global pointwise map

From HOT or HOT-POT, we may again compute a global point matching. Let $N_{\mathrm{tot}} = \sum_{i=1}^N N_i$ and

```
Algorithm 1: Hierarchical Object Matching
Input: Sets of $N$ source objects $\boldsymbol{X}$ and
       $M$ target objects $\boldsymbol{Y}$
       Cost function $c$ between points in $\mathbb{P}^2$
Output: Binary matching matrix $\Pi^{\mathrm{obj}}$
foreach $i \in [N]$, $j \in [M]$ do
    Compute pointwise POT cost
      $c^{\mathrm{obj}}(X_i, Y_j) = \mathrm{POT}_c(X_i, Y_j)$
Compute transport plan $\Pi^{\mathrm{obj}}$ minimizing
  $\mathrm{POT}_{c^{\mathrm{obj}}}(\boldsymbol{X}, \boldsymbol{Y})$
return $\Pi^{\mathrm{obj}}$
```

$M_{\mathrm{tot}} = \sum_{j=1}^{M} M_j$, and denote by $\Pi^{i,j} \in \mathbb{R}^{N_i \times M_j}$ a pointwise POT plan between X_i and Y_j, and by Π^{obj} the object-level plan. The global plan $\Pi^{\mathrm{glob}} \in \mathbb{R}_{\geq 0}^{N_{\mathrm{tot}} \times M_{\mathrm{tot}}}$ is obtained by embedding each local plan $\Pi^{i,j}$ into its corresponding block and scaling it by the transported mass $\Pi^{\mathrm{obj}}_{i,j}$: We set

$$\Pi^{\mathrm{glob}}_{(i,r),(j,s)} \coloneqq \Pi^{\mathrm{obj}}_{i,j}\, \Pi^{i,j}_{r,s}, \tag{17}$$

as the mass transported between x^i_r and y^j_s. Our resulting pointwise plan Π^{glob} becomes binary if all input plans are binary.

4 Algorithmic Considerations

In this section, we discuss several aspects of the implementation of our algorithms as well as error metrics. We will deal both with pointwise and objectwise matching.

4.1 OT Algorithms

OT. For computing $\mathrm{OT}_c(X, Y)$ with $c \in \{d^{\mathrm{epi}}, d^{\mathrm{ray}}, d^{\mathrm{reg}}_{\beta,\gamma}\}$, we apply the Earth Mover's Distance algorithm [8] implemented in the PythonOT library [20]. It returns a permutation matrix, but does not guarantee uniqueness and can be sensitive to input order.

POT. For partial OT, use the solver `ot.partial.partial_wasserstein` [11, 13] from PythonOT, which implements a relaxed optimal transport formulation, with a partial mass constraint $m = \min\{N, M\}/\max\{N, M\}$.

HOT/HOT-POT. For our hierarchical matching procedure, we utilize Algorithm 1 based on the aforementioned PythonOT solvers.

Remark 4.1 (Projecting onto Binary Matrices)**.** *While we know that our OT and POT problems can be solved by scaled (partial) permutation matrices, see [49, Prop. 2.1] and [3, Thm. 4.1], practical solvers relying on continuous relaxations may return soft transport plans. In that case, we project them to binary matrices by assigning each point to the maximizing index only if at least half of the mass is concentrated there, and discarding it otherwise.*

Remark 4.2 (Naive Matching)**.** *As baseline for comparing the performance of the OT and POT algorithms we use a* ***naive matching*** *procedure between two sets of points $X = \{x^1, \ldots, x^N\}$ and $Y = \{y^1, \ldots, y^M\}$. We find the smallest value within the cost matrix $C_{ij} = c(x^i, y^j)$, take the respective indices i^*, j^* for our matching, and remove the row i^* and the column j^*. As the smallest value might be non-unique, we take the first occurrence. We repeat this procedure until there is no row or column is left and obtain in total* $\min(N, M)$ *matches.*

4.2 Evaluation Criteria

Comparing with Ground-Truth Matching

Given a point cloud imaged by two different cameras, we can directly calculate the **pointwise mismatch rate** as the number of incorrectly matched point pairs divided by the total number of point pairs:

$$\frac{\#\text{incorrect point matches}}{\min\{N,\, M\}}.$$

For our parameter $m = \min\{N, M\}/\max\{N, M\}$, the total number of matches is $\min\{N, M\}$.

If we have multiple objects and each is described by a point cloud in the left and a point cloud in the right camera, we can perform object matching via HOT or HOT-POT. We calculate the **objectwise mismatch rate** as the number of incorrectly matched object pairs divided by the total number of object pairs:

$$\frac{\#\text{incorrect object matches}}{\min\{N, M\}}.$$

In our multi-modal landmarking system, each object is a single face that is described by i) the landmarks of an RGB tracker on the left and ii) a thermal landmarker on the right camera. For example, if the RGB tracker detects 5 faces and the thermal one detects 4, the total number of object pairs becomes 4. If our algorithm matches 2 face pairs correctly, the object mismatch ratio becomes $2/4 = 0.5$.

Evaluating Matching based on 3D Reconstruction.

In some applications, we might be more interested in reconstructing the true 3D point cloud than in recovering the exact point-to-point matching. In other scenarios, we might want to evaluate the pointwise matching quality in the absence of a ground-truth correspondence, e.g., due to occlusions. In this setting, an object is represented by two incompatible 3D point clouds, one visible on the left camera and one on the right.

In both cases, we can evaluate this setup based on the 3D reconstruction of our 3D point clouds. Given a pair $(x, y) \in \mathbb{P}^2 \times \mathbb{P}^2$, we can solve (1) for $w \in \mathbb{R}^3$ if we know the ground truth camera parameters and $r_x \times r_y \neq 0$, see Prop. 2.2. Thus, we can reconstruct (triangulate) a 3D point cloud and compare the **(squared) Wasserstein-2 distance** [49] between some ground truth point cloud and the reconstructed point cloud, i.e., we compute the minimum in (13) with the squared Euclidean distance in $\mathbb{R}^3$ as our cost function c (up to rescaling).

5 Numerical Experiments

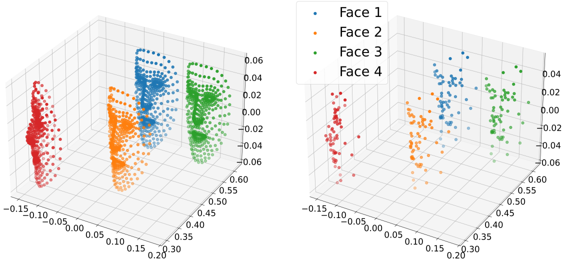
In our numerical experiments, we first perform experiments on synthetic data to allow for a quantitative comparison between the ray and the epipolar distance. Afterwards, we extend our analysis to real-world landmarking data.

5.1 Synthetic Faces Dataset

Dataset

We use an artificially created dataset of 3D points from four human faces, see Figure 3. The full dataset contains 1872 points corresponding to four 3D faces, each composed of the 468 landmarks of the MediaPipe canonical face model [44], see Figure 3a. The subsampled dataset consists of 65 points per face, corresponding to averaged landmarks of the 3D faces, see Figure 3b. For that purpose, the 468 landmarks are downsampled to 65 points by partitioning each region's sorted vertex indices into fixed numbers of chunks and taking the 3D centroid of each chunk. We know the ground truth correspondences, meaning that for each point in the left camera, the corresponding point in the right camera is known. Moreover, we have access to four distinct face labels as employed in our HOT formulation (15)–(16).

The camera projections x and y computed via the model (3) are shown in Figure 4. In particular, note that the projected faces partially overlap. Indeed, such scenarios may appear in practice since some trackers predict landmarks for occluded face regions via interpolating the face geometry [19] or motion in videos [65].



(a) Full (1872 pts) (b) Subsampled (260 pts)

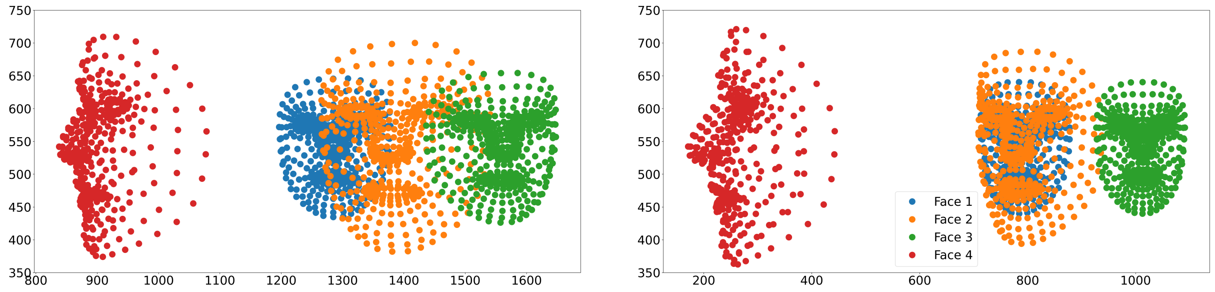
Fig. 3: 3D landmarks w of four faces.

c	Full ($N = 1872$)			Sub ($N = 260$)		
	Naive	OT	HOT	Naive	OT	HOT
d^{epi}	662 35%	681 36%	406 22%	66 25%	64 25%	0 0%
d^{ray}	809 43%	284 15%	264 14%	96 37%	0 0%	0 0%

Table 1: Point mismatch counts and ratios for the synthetic faces dataset.

Point Matching via OT and HOT with Ground Truth

We start by performing pointwise matching between the landmarks using the OT formulation (13) without any face labels. Table 1 reports the



(a) Left camera (points x_i)

(b) Right camera (points y_j)

Fig. 4: 2D camera images of the four faces dataset from Figure 3a.

mismatch rates of the OT matching and the naive matching in Remark 4.2 for our two distances. Overall, the results with the epipolar distance (11) are worse than with the ray distance (7). For the latter, OT leads to a considerable improvement over naive matching, reducing the number of errors from 809 to 284 for the full dataset, and from 96 to 0 for the subsampled data. Geometrically, the poor performance with the epipolar distance can be explained by the fact that points from *face 3* lie very close to epipolar lines (10) corresponding to *face 1*, see Figure 5.

If we include the face labels and employ the induced point-to-point HOT transport plan (17), we see a drastic improvement for the epipolar-based matching in Table 1. Nevertheless, the ray distance still leads to better results.

Figure 6 shows the cost matrices. For the epipolar distance, the cost nearly vanishes not only along the diagonal, but also along two sub-diagonals corresponding to the association between face 1 and face 3. In contrast, the ray distance can distinguish faces 1 and 3 more effectively.

Point Matching without Ground-Truth via POT

Next, we investigate the matching of all 1872 on the left camera and the subsampled 260 points on the right camera to assess the partial matching approach. Using the epipolar distance, 27.3 % of all points are matched to the wrong face. For the ray distance, this percentage goes down to 12.3 %. Following Section 4.2, the squared Wasserstein-2 distance between the reconstructed 3D point clouds is reported in Table 2.

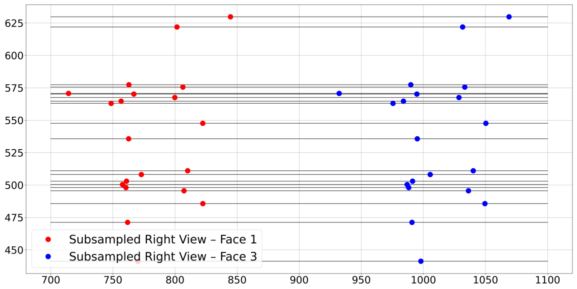
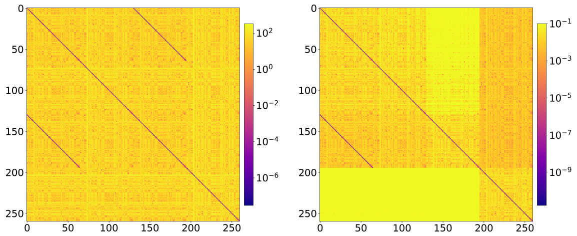


Fig. 5: Projection of 20 points from synthetic faces 1 (red) and 3 (blue) onto the right camera and epipolar lines of face 1, illustrating the difficulty of distinguishing faces with d^{epi} when two points are on a line.



(a) Cost: d^{epi}

(b) Cost: d^{ray}

Fig. 6: Cost matrices $[d(x_i, y_j)]_{ij}$ for the two geometric distances on the subsampled synthetic faces from Figure 3b. The epipolar cost exhibits low off-diagonal entries linking face 1 and face 3, while the ray cost suppresses these cross-face connections.

Face Matching via HOT-POT

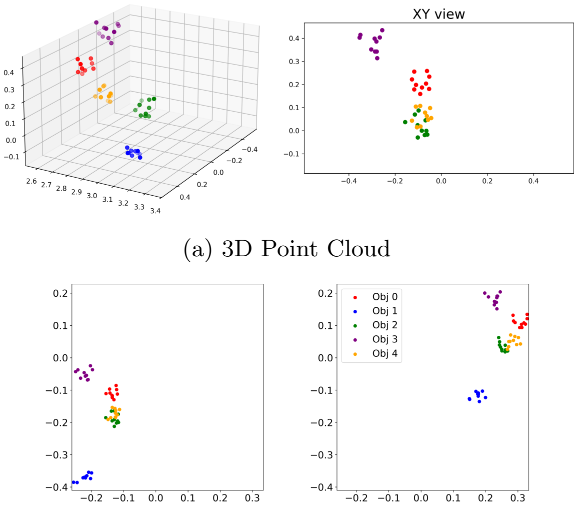
Lastly, we extend the partial matching comparison to object matching via HOT-POT by using the face labels for all points on one camera and

c	Full → Sub	Sub → Full
d^{epi}	1.39	1.45
d^{ray}	0.75	0.13

Table 2: Squared Wasserstein-2 distance for the synthetic faces dataset with POT, lower is better.

the subsampled points on the other camera. With this approach, the object-wise mismatch rates are 0 % for both distances, i.e., all faces are correctly matched. This holds for both setups, i.e., with the subsampled data on the left and the full data on the right and vice versa.

5.2 Synthetic Spheres Simulation



(a) 3D Point Cloud

(b) Left Camera

(c) Right Camera

Fig. 7: Spheres simulation with $\sigma = 0.005$. (a): Original data visualized as a 3D scatter plot and as a 2D projection. (b): Projection onto the left camera. (c): Projection onto the right camera. Shared axes of the camera planes in (b) and (c) highlight camera translation and rotation.

Simulation

We generate 100 synthetic 3D scenes composed of $N = 5$ disjoint spherical objects. For each object $k \in [5]$ we sample a center $c_k \in [-0.5, 0.5]^2 \times [2.5, 3.5]$ uniformly at random. The radius r_k of each sphere is drawn from a uniform distribution over $[0.05, 0.1]$. We reject and resample any proposed center whose sphere would overlap with an already placed sphere. On each sphere, we uniformly sample 10 points. This yields a point cloud of 50 points grouped into 5 spatially separated objects for each synthetic scene. We project all point clouds onto two cameras located at $(0, 0, 0)$ and $(1, 0, 0)$, set K as the identity matrix, and employ random camera rotations up to ±15 degrees to both. By Remark 2.4, camera rotations do not impact the ray distance. We repeat all experiments with varying levels of independent Gaussian noise $\mathcal{N}(0, \sigma^2)$ added to each point in both projections. An example is visualized in Figure 7.

Point Matching via OT and HOT

We investigate the point-to-point matching quality for all combinations of distances and pointwise matching, i.e., for combinations of (7), the depth-regularized ray distance (8) ($\gamma_1 = 2.5$, $\gamma_2 = 3.5$, $\beta = 10$), and the epipolar distance (11) with the naive, OT (13), and HOT matching (17). As described in Section 4.2, we evaluate our matching for different noise levels based on the mismatch ratio in Table 3 and the Wasserstein distance between the ground truth 3D point cloud and the reconstruction in Table 4.

Overall, we observe a quick deterioration in matching quality for increasing noise. The ray distance and the epipolar distance give comparable results in terms of the mismatch ratio, but we see an advantage of the ray distance in terms of the Wasserstein evaluation. The depth-regularized distance leads to a consistent improvement in the presence of noise, especially for the resulting 3D reconstruction. We see a clear advantage of the OT over the naive matching, with an additional performance boost via the HOT approach.

Object Matching via HOT

Using the HOT approach, we further investigate the resulting sphere-to-sphere matching in Table 5. Here, we obtain stable matching even in the presence of noise. Overall, we get the best results with the ray distance.

Method	$\sigma = 0.0$	0.001	0.005	0.01	0.05
d^{epi} Naive	0.0±0	44.1±11	80.6±8	88.5±6	95.9±3
d^{epi} OT	0.0±0	35.9±10	78.4±7	87.8±5	95.7±3
d^{epi} HOT	0.0±0	29.0±9	71.9±8	83.6±6	91.2±5
d^{ray} Naive	0.0±0	44.5±11	80.5±8	88.7±5	95.9±3
d^{ray} OT	0.0±0	36.1±11	76.9±8	86.9±5	95.5±3
d^{ray} HOT	0.0±0	29.5±9	71.4±8	82.6±6	91.3±5
d^{reg} Naive	0.3±1	37.5±11	75.7±8	85.2±6	94.3±4
d^{reg} OT	0.3±1	27.3±11	68.8±10	82.0±6	93.4±4
d^{reg} HOT	0.2±1	24.9±10	66.4±9	80.2±6	90.8±5

Table 3: Pointwise mismatch ratio for simulated spheres (in %, lower is better ↓).

Method	$\sigma = 0.0$	0.001	0.005	0.01	0.05
d^{epi} Naive	0.0±0	1.2±5	2.0±8	2.3±6	12.3±29
d^{epi} OT	0.0±0	1.1±5	1.6±4	2.1±5	8.5±15
d^{epi} HOT	0.0±0	0.1±0	0.1±0	0.4±2	5.5±18
d^{ray} Naive	0.0±0	1.2±5	1.4±3	2.1±5	7.8±16
d^{ray} OT	0.0±0	0.9±4	1.1±3	1.1±2	4.0±12
d^{ray} HOT	0.0±0	0.1±0	0.1±0	0.2±0	0.8±0
d^{reg} Naive	0.0±0	0.1±0	0.2±0	0.3±0	5.8±15
d^{reg} OT	0.0±0	0.1±0	0.1±0	0.1±0	0.6±2
d^{reg} HOT	0.0±0	0.1±0	0.1±0	0.1±0	0.6±1

Table 4: Squared Wasserstein-2 for 3D reconstruction of simulated spheres, where lower is better (↓).

c	$\sigma = 0.0$	0.001	0.005	0.01	0.05
d^{epi}	0.0±0	0.0±0	0.8±6	3.2±11	19.2±24
d^{ray}	0.0±0	0.0±0	0.4±4	2.0±9	10.8±19
d^{reg}	0.0±0	0.0±0	0.8±6	1.2±7	15.2±22

Table 5: Object mismatch using HOT for simulated spheres (in %, ↓).

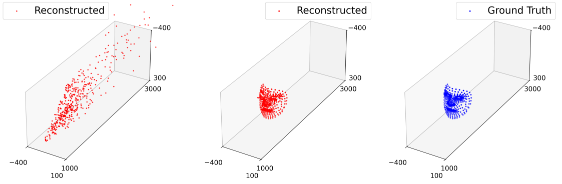
5.3 Matching RGB and Thermal Landmarks

Dataset

Our setup consists of two calibrated cameras with known intrinsic and extrinsic parameters, capturing frontal views of a human subject, see Figure 1. The calibrated images were obtained during a study at Saarland University, see [21, 22]. We consider the 468 Mediapipe landmarks [44] based on the first RGB camera as our left point cloud. For the right point cloud, we consider either the Mediapipe landmarks on the second RGB camera (“RGB-RGB”) or 5/70/478 landmarks from the thermal camera (“RGB-Thermal”), obtained via the landmarkers from [23] and [36].

RGB-RGB Point Matching via OT

In the RGB-RGB setup, we aim to match 468 Mediapipe landmarks with known ground-truth correspondence. Unlike the synthetic faces from Section 5.1, our calibration parameters and our landmark projections are subject to real-world noise. As a result, our OT matching based on the ray distance (7) leads to points being matched at practically infinite distance from the cameras. This highlights the advantage of the regularized ray distance (8), where we employ the parameters $\gamma_1 = 1550$, $\gamma_2 = 1750$ penalizing the depth of the scene, and the regularization strength parameter $\beta = 100$. The 3D reconstructions with and without regularization are shown in Figure 8. While the unregularized ray distance results in a poor reconstruction, the regularized distance reconstructs the shape of the face.

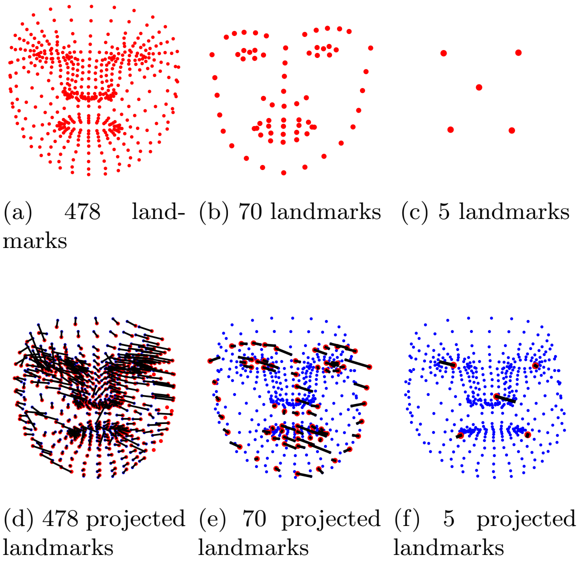


(a) d^{ray} (b) d^{reg} (c) GT

Fig. 8: 3D reconstruction of a face using matched Mediapipe landmarks (a) with the ray distance, (b) with the depth-regularized ray distance and (c) the ground truth (GT). While the first one gives poor results, regularization prevents matching with unreasonable depth.

RGB-Thermal Point Matching via POT

For the matching of RGB and thermal landmarks, there are no true correspondences, and a one-to-one matching is not possible for the 478-point (Fig. 9a), the 70-point (Fig. 9b), or the 5-point convention (Fig. 9c). We perform a partial matching between the different landmark conventions using the POT (14) with $m = \frac{\min\{N,M\}}{\max\{N,M\}}$ and the regularized ray distance ($\gamma_1 = 1550$, $\gamma_2 = 1750$, $\beta = 100$). We visualize the results in Figure 9 by projecting the thermal landmarks onto the RGB camera plane using the known calibration parameters and connecting matched points. Qualitatively, we see good matching with corresponding facial keypoints paired correctly across modalities.


(a) 478 landmarks (b) 70 landmarks (c) 5 landmarks

(d) 478 projected landmarks (e) 70 projected landmarks (f) 5 projected landmarks

Fig. 9: Top row: 2D right thermal camera images for different thermal landmark conventions. Bottom row: projection of matched thermal landmarks (red) onto the left RGB camera image. Light blue lines link thermal landmarks to the matched RGB landmarks (blue).

5.4 Matching RGB and Thermal Faces

Dataset

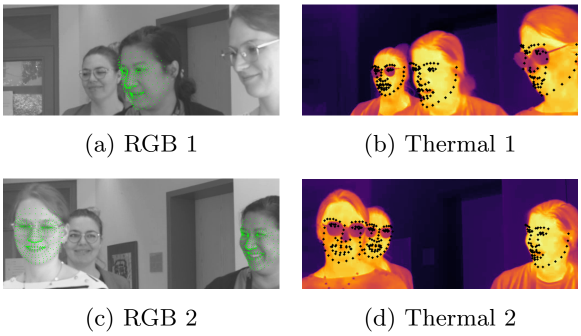
Finally, we evaluate our HOT-POT approach for cross-modal face matching with various real-world measurements. We use an RGB–thermal video recorded by the Systems Neuroscience and Neurotechnology Unit (SNNU) at Saarland University, showing three persons moving around a room. We extract 20 frame pairs from the videos and detect 468 RGB landmarks per face using Mediapipe [44] and 70 thermal landmarks using the T-FAKE landmaker [23] in combination with the TFW face tracker [36]. Notably, some faces are occluded in some frames, and the Mediapipe landmarker does not always detect every face, leading to a varying number of faces per frame and per camera. Here, the frames were chosen such that the time difference is small (both cameras have a different frame rate), and to ensure that each camera detects at least one face and at least one camera detects more than one face.

The camera calibration parameters are estimated from 59 calibration frames provided by SNNU, showing an asymmetric circle-grid target observed at different positions and orientations. Calibration is performed using standard routines from OpenCV [9].

Examples are visualized in Figure 10. There are various sources of errors resulting from i) the estimated camera calibration, ii) the inconsistent number of landmarks and faces between both cameras, iii) temporal delays between the two camera frames, and iv) landmark detection errors.

Face Matching via HOT-POT

Based on the estimated camera calibration parameters, we run our HOT-POT algorithm for face matching using the ray distance (7), the depth-regularized ray distance (8) with $\gamma_1 = 500$, $\gamma_2 = 5000$, and $\beta = 1$, as well as the epipolar distance (11).


(a) RGB 1 (b) Thermal 1

(c) RGB 2 (d) Thermal 2

Fig. 10: Two cropped RGB and thermal landmark pairs extracted from our video provided by SNNU. For visualization, RGB images are shown in grayscale. Between RGB and thermal camera, there are small time lags and the number of detected faces may differ.

Frames are marked as correct if all landmarked faces are correctly matched and the mismatch rate is averaged over 20 frame pairs. As shown in Table 6, the epipolar distance d_{epi} incorrectly matches 24% of all available face pairs, resulting in errors in 5 out of 20 frames. In contrast, the ray distance d_{ray} achieves a mismatch rate of 5%, with only a single erroneous frame. The depth-regularized ray distance d_{reg} performs best, correctly matching all pairs of faces in all 20 frames.

Method	Correct Frames	Mismatch rate (%)
d_{ray}	19/20	5%
d_{reg}	20/20	0%
d_{epi}	15/20	24%

Table 6: Comparison of distance metrics for the real RGB-thermal data.

6 Conclusions

We proposed a new approach for 3D stereo matching of sparse point clouds using a partial OT framework, where the matching costs are derived from epipolar geometry. While our first cost is based on the 3D distance between the rays through the camera plane and the focal point, our second cost relies on enforcing the epipolar constraints. The ray-based cost, combined with a regularization term, provides more robust performance than the commonly used epipolar constraint-based cost, especially in noisy settings. For matching objects rather than single points, we developed a hierarchical matching framework, which first solves the POT between all possible object pairs and then calculates the matching among the objects. While we found a large sensitivity to measurement noise for the pointwise approach, the HOT matches the objects correctly in the case of large deviations and real-world measurements.

In the future, we want to extend our method to perform a three-way matching via multimarginal OT [2, 41, 48], integrate keypoint features such as color, and incorporate our methods into dense stereo matching algorithms such as H-Net [29]. Our application may become useful for public health screening, where one is interested in identifying persons with elevated temperature to prevent the spreading of infectious diseases.

Acknowledgements. AC gratefully acknowledges funding from the Berlin Mathematical School and support from the French Research Agency through the Holibrain project (ANR-23-CE45-0020). MQ gratefully acknowledges funding by the German Research Foundation (DFG): STE 571/19-1, project number 495365311, within the Austrian Science Fund (FWF) SFB 10.55776/F68 “Tomography Across the Scales”. MP gratefully acknowledges funding from the German Research Foundation (DFG) within the project BIOQIC (GRK2260/289347353). PF acknowledges support from the Japan Society for the Promotion of Science (JSPS) under Grant No. PE25715. GK acknowledges funding by the BMBF VI-ScreenPRO (ID: 100715327). For open access purposes, the author has applied a CC BY public copyright license to any author-accepted manuscript version arising from this submission.

We are especially grateful to Mayur Bhamborae of the Systems Neuroscience and Neurotechnology Unit (SNNU) at Saarland University for providing the calibration data and frames, as well as the real RGB–thermal data used in our numerical experiments and to Daniel J. Strauss for enabling the scientific exchange between Saarland University and TU Berlin.

References

[1] D. Alvarez-Melis and N. Fusi. Geometric dataset distances via optimal transport. In *Advances in Neural Information Processing Systems*, pages 21428–21439. Curran Associates, 2020. Article No.: 1799.

[2] F. A. Ba and M. Quellmalz. Accelerating the Sinkhorn algorithm for sparse multi-marginal optimal transport via fast Fourier transforms. *Algorithms*, 15(9):311, 2022.

[3] Y. Bai, B. Schmitzer, M. Thorpe, and S. Kolouri. Sliced optimal partial transport. In *Proceedings of the CVPR'23*, pages 13681–13690. IEEE, 2023.

[4] F. Beier, J. von Lindheim, S. Neumayer, and G. Steidl. Unbalanced multi-marginal optimal transport. *Journal of Mathematical Imaging and Vision*, 65(3):394–413, 2023.

[5] M. Benning and M. Burger. Modern regularization methods for inverse problems. *Acta Numerica*, 27:1–111, 2018.

[6] C. Bonet, C. Vauthier, and A. Korba. Flowing datasets with Wasserstein over Wasserstein gradient flows. In *Proceedings of the ICML'25*, 2025.

[7] N. Bonneel, J. Rabin, G. Peyré, and H. Pfister. Sliced and Radon Wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51(1):22–45, 2015.

[8] N. Bonneel, M. van de Panne, S. Paris, and W. Heidrich. Displacement interpolation using Lagrangian mass transport. *ACM Transactions on Graphics*, 30(6):158:1–158:12, 2011.

[9] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[10] A. Bulat and G. Tzimiropoulos. How far are we from solving the 2D & 3D face alignment problem? (and a dataset of 230,000 3D facial landmarks). In *Proceedings of the ICCV'17*, pages 1021–1030, 2017.

[11] L. A. Caffarelli and R. J. McCann. Free boundaries in optimal transport and Monge-Ampère obstacle problems. *Annals of Mathematics*, 171(2):673–730, 2010.

[12] X. Cai, J. H. Fitschen, M. Nikolova, G. Steidl, and M. Storath. Disparity and optical flow partitioning using extended potts priors. *Information and Inference: A Journal of the IMA*, 4(1):43–62, 2015.

[13] L. Chapel, M. Alaya, and G. Gasso. Partial optimal transport with applications on positive-unlabeled learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 15426–15437. Curran Associates, 2020.

[14] L. Chizat, G. Peyré, B. Schmitzer, and F.-X. Vialard. Unbalanced optimal transport: Dynamic and Kantorovich formulations. *Journal of Functional Analysis*, 274(11):3090–3123, 2018.

[15] M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, volume 26, pages 2292–2300. Curran Associates, 2013.

[16] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the CVPR'05*, pages 886–893. IEEE, 2005.

[17] F. Darmon and P. Monasse. The polar epipolar rectification. *Image Processing On Line*, 11:56–75, 2021.

[18] J. Delon and A. Desolneux. A Wasserstein-type distance in the space of Gaussian mixture models. *SIAM Journal on Imaging Sciences*, 13(2):936–970, 2020.

[19] J. Deng, A. Roussos, G. Chrysos, E. Ververas, I. Kotsia, J. Shen, and S. Zafeiriou. The Menpo benchmark for multi-pose 2D and 3D facial landmark localisation and tracking. *International Journal of Computer Vision*, 127(6):599–624, 2019.

[20] R. Flamary, N. Courty, A. Gramfort, M. Z. Alaya, A. Boisbunon, S. Chambon, L. Chapel, A. Corenflos, K. Fatras, N. Fournier, L. Gautheron, N. T. Gayraud, H. Janati, A. Rakotomamonjy, I. Redko, A. Rolet, A. Schutz, V. Seguy, D. J. Sutherland, R. Tavenard, A. Tong, and T. Vayer. POT: Python Optimal Transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021.

[21] P. Flotho, M. J. Bhamborae, T. Grün, C. Trenado, D. Thinnes, D. Limbach, and D. J. Strauss. Multimodal data acquisition at SARS-CoV-2 drive through screening centers: Setup description and experiences in Saarland, Germany. *Journal of Biophotonics*, 14(8):e202000512, 2021.

[22] P. Flotho, C. Heiss, G. Steidl, and D. J. Strauss. Lagrangian motion magnification with double sparse optical flow decomposition. *Frontiers in Applied Mathematics and Statistics*, 9:1164491, 2023.

[23] P. Flotho, M. Piening, A. Kukleva, and G. Steidl. T-FAKE: Synthesizing thermal images for facial landmarking. In *Proceedings of the CVPR'25*, pages 26356–26366. IEEE, 2025.

[24] H. Fsian, V. Mohammadi, P. Gouton, and S. Minaei. Comparison of stereo matching algorithms for the development of disparity map, 2022. arXiv preprint arXiv:2210.15926.

[25] M. Galeotti, A. Sarti, and G. Citti. A framework for stereo vision via optimal transport, 2022. arXiv preprint arXiv:2207.00333.

[26] Y. Guo, X. Qi, J. Xie, C.-Z. Xu, and H. Kong. Unsupervised cross-spectrum depth estimation by visible-light and thermal cameras. *IEEE Transactions on Intelligent Transportation Systems*, 24(10):10937–10947, 2023.

[27] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2nd edition, 2004.

[28] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, UK, 2 edition, 2004.

[29] B. Huang, J.-Q. Zheng, S. Giannarou, and D. S. Elson. H-net: Unsupervised attention-based stereo depth estimation leveraging epipolar geometry. In *Proceedings of the CVPR'22*, pages 4460–4467. IEEE, 2022.

[30] K. Häming and G. Peters. Extension of the generalized image rectification - catching the infinity cases. In *Proceedings of the ICINCO'07*, pages 275–279. AAAI, 2007.

[31] I. Kemelmacher-Shlizerman, S. M. Seitz, D. Miller, and E. Brossard. The MegaFace benchmark: 1 million faces for recognition at scale. In *Proceedings of the CVPR'16*, pages 4873–4882. IEEE, 2016.

[32] S. Kim, D. Min, B. Ham, M. N. Do, and K. Sohn. DASC: Robust dense descriptor for multi-modal and multi-spectral correspondence estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9):1712–1729, 2016.

[33] P. A. Knight. The Sinkhorn–Knopp algorithm: convergence and applications. *SIAM Journal on Matrix Analysis and Applications*, 30(1):261–275, 2008.

[34] K. Y. Kok and P. Rajendran. A review on stereo vision algorithm: Challenges and solutions. *Proceedings of the ECTI-CIT'19*, 13(2):112–128, 2019.

[35] S. Kolouri, K. Nadjahi, U. Simsekli, R. Badeau, and G. K. Rohde. Generalized sliced Wasserstein distances. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, 2019.

[36] A. Kuzdeuov, D. Aubakirova, D. Koishigarina, and H. A. Varol. TFW: Annotated thermal faces in the wild dataset. *IEEE Transactions on Information Forensics and Security*, 17:2084–2094, 2022.

[37] R. Leroy, P. Trouvé-Peloux, F. Champagnat, B. Le Saux, and M. Carvalho. Pix2Point: Learning outdoor 3d using sparse point clouds and optimal transport. In *Proceedings of the MVA'21*, pages 1–5. IEEE, 2021.

[38] K. Li, L. Wang, Y. Zhang, K. Xue, S. Zhou, and Y. Guo. LoS: Local structure-guided stereo matching. In *Proceedings of the CVPR'24*, pages 19746–19756, 2024.

[39] R. Li, G. Lin, and L. Xie. Self-point-flow: Self-supervised scene flow estimation from point clouds with optimal transport and random walk. In *Proceedings of the CVPR'21*, pages 15572–15581. IEEE, 2021.

[40] X. Liang and C. Jung. Deep cross spectral stereo matching using multi-spectral image fusion. *IEEE Robotics and Automation Letters*, 7(2):5373–5380, 2022.

[41] T. Lin, N. Ho, M. Cuturi, and M. I. Jordan. On the complexity of approximating multimarginal optimal transport. *Journal of Machine Learning Research*, 23:1–43, 2022.

[42] C.-W. Liu, H. Wang, S. Guo, M. J. Bocus, Q. Chen, and R. Fan. *Stereo Matching: Fundamentals, State-of-the-Art, and Existing Challenges*, pages 63–100. Springer, Singapore, 2023.

[43] Y. Liu, Y. Liu, S. Yan, C. Chen, J. Zhong, Y. Peng, and M. Zhang. A multi-view thermal–visible image dataset for cross-spectral matching. *Remote Sensing*, 15(1):174, 2022.

[44] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. Yong, J. Lee, et al. Mediapipe: A framework for perceiving and processing reality. In *Proceedings of the Workshop on Computer Vision for AR/VR at CVPR'19*, 2019.

[45] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors.

IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(10):1615–1630, 2005.

[46] K. Nguyen. An introduction to sliced optimal transport: Foundations, advances, extensions, and applications. *Foundations and Trends in Computer Graphics and Vision*, 17(3-4):171–406, 2025.

[47] K. Nguyen, H. Nguyen, T. Pham, and N. Ho. Lightspeed geometric dataset distance via sliced optimal transport. In *Proceedings of the ICML'25*. OpenReview.net, 2025.

[48] B. Pass. Multi-marginal optimal transport: Theory and applications. *ESAIM: Mathematical Modelling and Numerical Analysis*, 49(6):1771–1790, 2015.

[49] G. Peyré and M. Cuturi. Computational Optimal Transport. *Foundations and Trends in Machine Learning*, 11(5–6):355–607, 2019.

[50] M. Piening and R. Beinert. Slicing the Gaussian mixture Wasserstein distance. *Transactions on Machine Learning Research*, 2025.

[51] M. Piening and R. Beinert. Slicing Wasserstein over Wasserstein via functional optimal transport, 2025. arXiv preprint arXiv:2509.22138.

[52] P. Pinggera, T. Breckon, and H. Bischof. On cross-spectral stereo matching using dense gradient features. In *Proceedings of the BMVC'12*, pages 103.1–103.12, 2012.

[53] M. Pollefeys, R. Koch, and L. Van Gool. A simple and efficient rectification method for general motion. In *Proceedings of the ICCV'99*, volume 1, pages 496–501 vol.1. IEEE, 1999.

[54] S. J. D. Prince. *Computer Vision: Models, Learning, and Inference.* Cambridge University Press, 1st edition, 2012.

[55] G. Puy, A. Boulch, and R. Marlet. FLOT: Scene flow on point clouds guided by optimal transport. In A. Vedaldi, H. Bischof, T. Brox, and J. M. Frahm, editors, *Proceedings of the ECCV'20*, volume 12373, pages 595–612. Springer, Cham, 2020.

[56] M. Quellmalz, R. Beinert, and G. Steidl. Sliced optimal transport on the sphere. *Inverse Problems*, 39(10):105005, 2023.

[57] M. Quellmalz, L. Buecher, and G. Steidl. Parallelly sliced optimal transport on spheres and on the rotation group. *Journal of Mathematical Imaging and Vision*, 66:951–976, 2024.

[58] D. Scharstein. *View Synthesis Using Stereo Vision*, volume 1586 of *Lecture Notes in Computer Science.* Springer Berlin, Heidelberg, 1999.

[59] K. Schauwecker, R. Klette, and A. Zell. A new feature detector and stereo matching method for accurate high-performance sparse stereo matching. In *Proceedings of the IROS'12*, pages 5171–5176. IEEE, 2012.

[60] B. Schmitzer and C. Schnörr. A hierarchical approach to optimal transport. In A. Kuijper, K. Bredies, T. Pock, and H. Bischof, editors, *Proceedings of the SSVM'13*, pages 452–464, Berlin, Heidelberg, 2013. Springer.

[61] T. Séjourné, G. Peyré, and F.-X. Vialard. Unbalanced optimal transport, from theory to numerics. *Handbook of Numerical Analysis*, 24:407–471, 2023.

[62] Z. Shen, J. Feydy, P. Liu, A. H. Curiale, R. San Jose Estepar, R. San Jose Estepar, and M. Niethammer. Accurate point cloud registration with robust optimal transport. *Advances in Neural Information Processing Systems*, 34:5373–5389, 2021.

[63] C. Steger and M. Ulrich. A multi-view camera model for line-scan cameras with telecentric lenses. *Journal of Mathematical Imaging and Vision*, 64(2):105–130, 2022.

[64] V. Stein, S. Neumayer, N. Rux, and G. Steidl. Wasserstein gradient flows for Moreau envelopes of f-divergences in reproducing kernel Hilbert spaces. *Analysis and Applications*, 2025.

[65] K. Sun, W. Wu, T. Liu, S. Yang, Q. Wang, Q. Zhou, Z. Ye, and C. Qian. FAB: A robust facial landmark detection framework for motion-blurred videos. In *Proceedings of the CVPR'19*, pages 5462–5471. IEEE, 2019.

[66] Y. Tang, H. Li, X. Sun, J.-M. Morvan, and L. Chen. Principal curvature measures estimation and application to 3d face recognition. *Journal of Mathematical Imaging and Vision*, 59(2):211–233, 2017.

[67] C. Villani. *Topics in Optimal Transportation.* American Mathematical Society, Providence, 2003.

[68] E. Wood, T. Baltrušaitis, C. Hewitt, S. Dziadzio, T. J. Cashman, and J. Shotton. Fake it till you make it: face analysis in the wild using synthetic data alone. In *Proceedings of the ICCV'21*, pages 3681–3691. IEEE, 2021.

[69] Y. Wu and Q. Ji. Facial landmark detection: A literature survey. *International Journal of Computer Vision*, 127(2):115–142, 2019.

[70] M. Yurochkin, S. Claici, E. Chien, F. Mirzazadeh, and J. M. Solomon. Hierarchical optimal transport for document representation. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32, pages 1599–1609. Curran Associates, 2019.

[71] S. Zhang, W. Su, F. Liu, and L. Sun. Review of stereo matching based on deep learning. *Displays*, 87:102940, 2025.

[72] T. Zhi, B. R. Pires, M. Hebert, and S. G. Narasimhan. Deep material-aware cross-spectral stereo matching. In *Proceedings of the CVPR'18*, pages 1916–1925, 2018.