scientific reports

Check for updates

OPEN

A principled representation of elongated structures using heatmaps

Florian Kordon[1,2,3]✉, Michael Stiglmayr[4], Andreas Maier[1,2], Celia Martín Vicario[1], Tobias Pertlwieser[1] & Holger Kunze[1,3]

The detection of elongated structures like lines or edges is an essential component in semantic image analysis. Classical approaches that rely on significant image gradients quickly reach their limits when the structure is context-dependent, amorphous, or not directly visible. This study introduces a principled mathematical description of elongated structures with various origins and shapes. Among others, it serves as an expressive operational description of target functions that can be well approximated by Convolutional Neural Networks. The nominal position of a curve and its positional uncertainty are encoded as a heatmap by convolving the curve distribution with a filter function. We propose a low-error approximation to the expensive numerical integration by evaluating a distance-dependent function, enabling a lightweight implementation with linear time complexity. We analyze the method's numerical approximation error and behavior for different curve types and signal-to-noise levels. Application to surgical 2D and 3D data, semantic boundary detection, skeletonization, and other related tasks demonstrate the method's versatility at low errors.

The detection of lines and contours is a central topic of image analysis and image understanding. Applications are widespread and range from computer vision, defect detection[1,2], facial boundary and expression recognition[3–6], geographic data analysis[7] to medical imaging[8]. The images may be photographs of streets where the curbside should be detected or images of wheel rims where cracks are identified[9,10]. Other applications consider satellite imagery to monitor the extent of glacial termini[11,12] or X-ray monitoring of a catheter during a surgical intervention[13]. While these applications differ in their nature, the underlying mathematical problem stays the same. A line, boundary, ridge, or contour—subsequently summarized by the term *elongated structure*—should be detected in an image.

Classical approaches typically tackle this task by using finite differences in intensity or color to approximate image gradients[14–17]. However, these methods fail when no common step representation can be found or if the noise level varies too much over the image. Active Contour Models[18] alleviate this problem by minimizing the energy of a contour spline subject to some additional constraints. While a suitable energy term can be automatically derived by evaluating the contour-surrounding structures[19,20], the fundamental problem of a good initialization remains, and these methods reach their limits when no shape model for the target structure can be found. This is especially the case for tasks in unconstrained settings like human pose estimation problems, where there is an infinite number of combinations of the person's poses and clothes with the background[21,22]. For such problems, Convolutional Neural Networks (CNN) have become state of the art[23]. By sequential aggregation of features, abstract representation models can be optimized that generalize well across a large number of image impressions and shape realizations.

However, until now, no default methodology has been established for the representation of elongated structures and the subsequent formulation of an optimization target. This is particularly the case for structures that are not directly visible, do not have a specific shape or form, or are defined by their surrounding, often varying context. Current literature proposes several different strategies to address this detection task (Supplementary Material Table S1).

[1]Pattern Recognition Lab, Friedrich-Alexander Universität Erlangen-Nürnberg, 91058 Erlangen, Germany. [2]Erlangen Graduate School in Advanced Optical Technologies (SAOT), Friedrich-Alexander Universität Erlangen-Nürnberg, 91052 Erlangen, Germany. [3]Advanced Therapies, Siemens Healthcare GmbH, 91301 Forchheim, Germany. [4]Optimization Group, Institute of Mathematical Modelling, Analysis and Computational Mathematics, University of Wuppertal, 42119 Wuppertal, Germany. ✉email: florian.kordon@fau.de

A typical approach is to formulate the optimization task as a *binary segmentation* problem, where the elongated structure is represented as a discretely sampled line of small width[24]. The algorithm is optimized to delineate the line pixels and all background using a (binary) cross-entropy (BCE) or soft dice (Dice) cost function. Unfortunately, segmentation inherently suffers from a substantial class imbalance if the target structure constitutes only a small foreground area. Also, label bias and minor offsets in the predicted segmentation strongly influence the optimization cost, rendering the optimization target volatile. As a result, the prediction is prone to fragmentation and gaps in ambiguous regions. One effective approach to mitigate this problem is a distance-weighted loss that relaxes the optimization target in the curve-near region[25,26]. Although this reduces fragmentation, it also smooths out potentially important true-positive gaps in the structure. Other approaches compose multiple segmentation cost functions[27] or leverage regional integrals to penalize errors in contour space, effectively circumventing large magnitude differences present in overlap-based cost functions[28].

A second approach is the extraction of the centerline/symmetry axis, often referred to as *skeletonization*. Traditional algorithms describe the centerline as the set of midpoints of maximally inscribed disks in an object[29]. The skeleton resulting from this definition is very sensitive to small boundary changes caused by noise or errors during boundary extraction[30]. Advanced algorithms alleviate these problems with costly post-processing[31], define the task as a learning problem in scale-space[32], utilize multi-task CNNs to combine skeleton pixel detection and skeleton scale regression[33,34], or encode the center-line as points with positive inward flux in a locally confined vector field[35]. While the latter method does not presume explicit boundary information, flux areas of multiple objects in a scene must not overlap, which limits the application to arbitrary elongated structures.

Approaches for facial boundary detection mostly build upon the idea of *heatmap-based landmark localization* where the landmarks' positional likelihood is encoded via spatial activation maps. These maps are typically computed by sampling a bivariate Gaussian function whose mean is at the ground truth coordinate[36,37]. By approximating a polyline that connects multiple structure-describing landmarks, this approach can be extended for elongated structures[22,23,38]. The polyline is mapped to a heatmap by either merging individual heatmap representations of the polyline points[3,39] or by resolving pixel-wise distances to the structure using a distance transform[5,6,40,41]. However, an inherent issue with these approaches is the piecewise linear effect when connecting distant landmarks, causing inaccuracy at high-curvature regions which inversely scales with the number of landmarks[4]. Despite this effect, such representation is desirable as it provides high spatial generalization during inference and promotes a stable gradient signal for every spatial position, reducing the amount of data needed. Furthermore, the choice of the heatmap-generating function is theoretically arbitrary, making it easy to adapt the type of representation to the specific task.

Similarities are found in recent applications of graph convolutional neural networks (GCNNs) to line, contour, and semantics analysis[42,43]. The relations and constraints between salient positions of an elongated structure can be represented by the adjacency matrix of a graph. This allows us to incorporate global semantics and overcome topological inconsistencies which might appear upon image domain shifts or information loss by artifacts and occlusion. To this end, the typical process is to first locate relevant landmarks using image features and then perform semantic linkage-based graph analysis with GCNNs to obtain plausible landmark positions[42,44,45].

In this work, we propose a mathematical description of elongated structures that offers a fundamental and holistic view of curves in space and provides a general encoding for a wide range of practical applications. This description shall serve, among others, as a target for optimization-based algorithms in which a neural network learns a representation of the elongated structures and approximates their distribution in space (Fig. 1). For these purposes, we adapt the concept of heatmap representation of single points to an arbitrary continuous curve in space with associated positional uncertainty. The distribution of curve points is convolved with a filter function, allowing prior domain knowledge of the expected signal and noise types to be included. To avoid numerical integration, we introduce a low-error approximation that simplifies the integral calculation to an evaluation of a distance-dependent function. To understand the representation's fundamental properties, we analyze the numerical approximation error, compare the method to several related spatial representation types, study the influence of different signal/heatmap configurations, and evaluate the susceptibility to noise (Sect. "Analysis of representation properties", Supplementary Material Sect. S2). The findings of this analysis are validated for various application examples in 2D and 3D (Fig. 1), coming with challenging and diverse types of elongated structures (Supplementary Material Table S2). For this purpose, the network is given a data sample as input and is tasked to approximate the structure's position by matching the heatmap obtained using the proposed heatmap representation. This matching is supervised using a divergence measure, e.g., mean squared error, between the network prediction and the generated target.

We highlight our contributions to pattern analysis research in the following areas:

- We introduce a general mathematical description for 2D and 3D curves of various origins, shapes, and information levels. A primary application is the use of this description as an optimization target in CNN-based representation learning.
- We provide a numerical solution that is easily configurable and allows tailoring to the observed or expected curve type, the uncertainty of the curve's position, and the application scenario. Unlike most conventional representation methods that rely on dense curve representations, this solution supports highly sparse curve signals with few sampled data points, completely avoiding the piecewise linear effect if desired.
- We study the numerical approximation error, showing negligible error magnitudes if no extreme curve curvatures exist. Furthermore, we draw clear lines of the proposed method to related spatial representations in the context of gradient-based optimization algorithms.
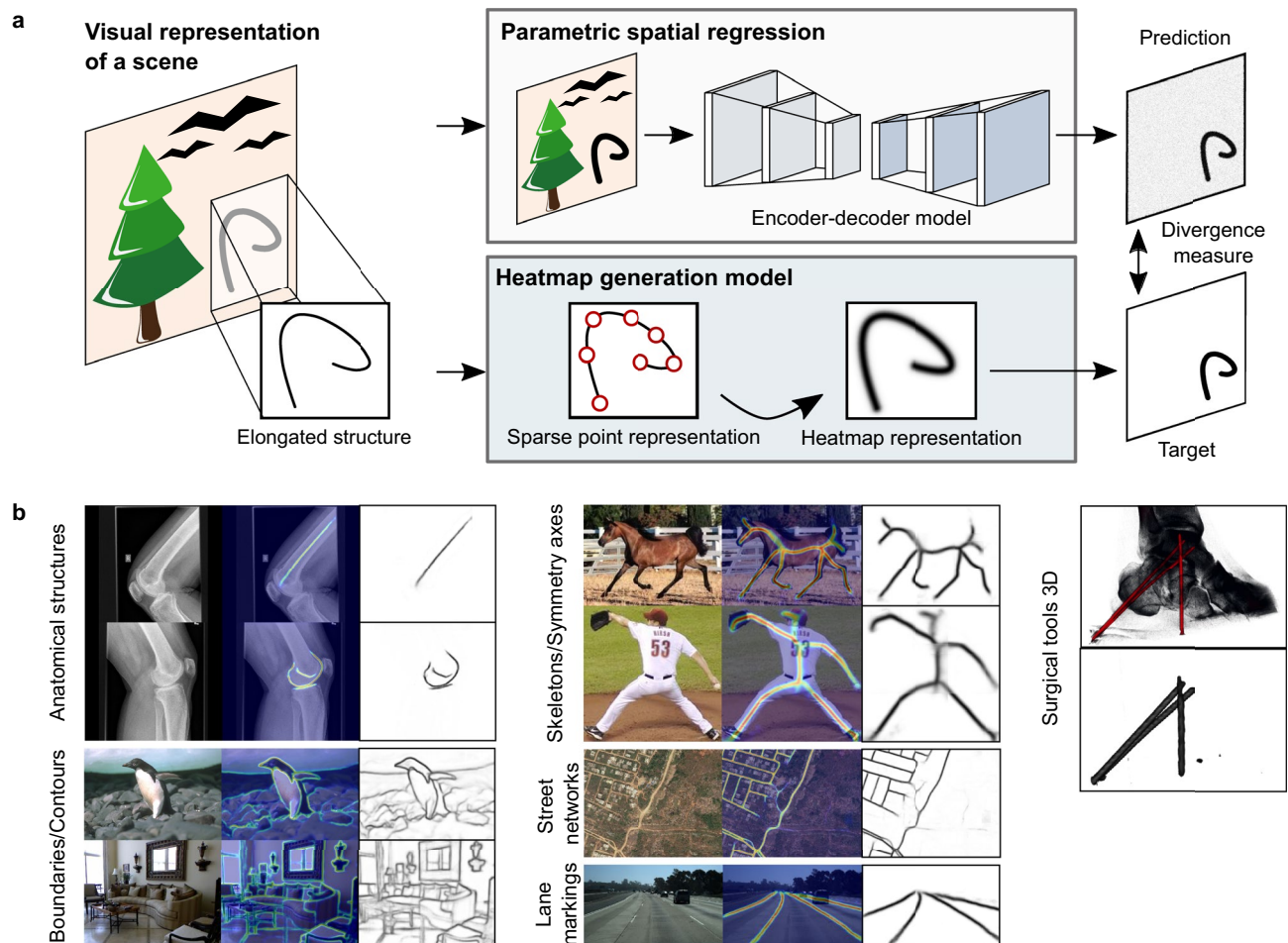
Figure 1. Conceptual overview and application examples. (**a**) Main use case of the heatmap representation as an operational description of the target function in a learning system. (**b**) application of the heatmap representation to anatomical feature detection, boundary detection[46,47], skeletonization[33,48,49], line detection[50], segmentation[51], and surgical tool detection in 3D (Sect. "Application to 2D and 3D representation problems", Supplementary Material Sects. S3–S6).

- We analyze the method's properties for different curve types and representations. We show that Gaussian-distributed heatmaps are generally applicable for arbitrary signals, and derive implications for practical applications.
- We demonstrate the representation's applicability as a target for CNN-based optimization for a broad array of tasks: anatomical feature detection (Sect. "Anatomical structure detection on knee radiographs"), 3D implant detection (Sect. "Detection of surgical implants in 3D CBCT volumes"), object contour detection (Supplementary Material Sect. S3), skeletonization (Supplementary Material Sect. S4), road lane detection (Supplementary Material Sect. S5), and road network segmentation (Supplementary Material Sect. S6).

Heatmaps for elongated structures

We start our considerations by looking at operational descriptions of target functions that a CNN-based learning system can adequately approximate. Fundamentally, the type and expressiveness of a representation ultimately define how well the target function, e.g., the distribution of a curve in space, can be approximated. They further dictate the number of data samples needed to train the learning system until it reaches a sufficient approximation fidelity[52]. On the other hand, the choice of representation, e.g., a spatial probability map or parametric encoding, influences which cost functions can be used. With these aspects in mind, we subsequently introduce a general and versatile mathematical description for elongated structures of different origins and shapes. We derive their representation as spatial heatmaps and propose a simple and fast implementation that allows for efficient network training.

From convolution to distance function. The representation of an elongated structure as a heatmap can be motivated from several perspectives. On the one hand, heatmaps model the positional uncertainty of the curve. Thus, the position of a curve $\tilde{\mathbf{c}}(t) = \mathbf{c}(t) + \mathbf{z}(t)$ with parameter variable $t \in [t_0, t_n]$ is the sum of the nominal position of the curve $\mathbf{c}(t)$ superposed by a noise term $\mathbf{z}(t)$ with probability density function **pdf**. By

evaluating the parameter variable t for the entire interval of definition from t_0 to t_n, it traces a noisy image of the curve in space. Moreover, a heatmap can be used as an optimization target for CNN training, where the correspondence with some ground truth is evaluated with a specific type of cost function. For the binary segmentation of an image into relevant curve points and background, BCE or Dice cost functions are used. If we can assume a normal distribution to model the curve point position with high likelihood, a least-squares-based cost is a natural choice.

Besides representing a curve explicitly as the image of a parameter-dependent function, it can also be described implicitly as a level set, which is defined by the (non-linear) equation

$$C = \left\{ \boldsymbol{x} \in \mathbb{R}^n : f(\boldsymbol{x}) = 0 \right\}. \tag{1}$$

The function $f(\boldsymbol{x})$ is a measure for the distance of a point $\boldsymbol{x}$ to the curve. In this generic setting, it would suffice to use a function $f(\boldsymbol{x})$ which evaluates to 0 at curve points. However, for simplicity of notation, we introduce f as a distance measure here, which will be required in later sections. Using this distance function, the distribution $\mathscr{D}$ of a curve in the space $\mathbb{R}^n$ is

$$\mathscr{D}(\boldsymbol{x}) = \delta\big(f(\boldsymbol{x})\big), \tag{2}$$

with $\delta(s)$, $s \in \mathbb{R}$ being the Dirac function. We can now convolve this distribution with a filter function $w(\boldsymbol{x})$ and obtain the heatmap

$$\begin{aligned} H(\boldsymbol{x}) &= \mathscr{D}(\boldsymbol{x}) * w(\boldsymbol{x}) \\ &= \int_{\mathbb{R}^2} \delta(f(\boldsymbol{y}))\, w(\boldsymbol{x} - \boldsymbol{y})\, \mathrm{d}\boldsymbol{y} \\ &= \int_C w(\boldsymbol{x} - \boldsymbol{y})\, \mathrm{d}\boldsymbol{y}. \end{aligned} \tag{3}$$

However, solving $H(\boldsymbol{x})$ requires the evaluation of the line integral along a potentially curved line through the filter kernel. Depending on the implementation, no closed-form solution is available, so that numerical integration is necessary. For the special cases of rotationally symmetric functions and straight lines, this could be implemented using look-up tables. However, since we are interested in a general description that also includes curved lines, we introduce a simplification for the description of $H(\boldsymbol{x})$, which is considerably faster than numerical integration: In the following, we assume that w is radially symmetric, i.e., $w(\boldsymbol{x}) = w(\boldsymbol{y})$ for all $\|\boldsymbol{x}\|_2 = \|\boldsymbol{y}\|_2$. If we assume that C is an infinite straight line, the evaluation of the integral simplifies to a function $h : \mathbb{R} \to \mathbb{R}$ which only depends on the distance to the line C:

$$H(\boldsymbol{x}) = h(f(\boldsymbol{x})). \tag{4}$$

In the following, we will use (4) also in the general case as an approximation for the integral in (3).

Numerical approximation and implementation. In this section, we describe an efficient implementation for this approximation of convolution by a distance-dependent function.

We assume that a finite set of data points can describe the original structure sufficiently well. These points are then used to derive a functional curve approximation by performing a (smooth) parametric B-spline interpolation. As only the spatial positions within a limited distance to the curve are numerically relevant for evaluating the distance function, we restrict calculations to a compact hull along the curve. Therefore, we sample the interpolating function and form a 2D polyline by connecting consecutive sample points (Fig. 2). For each resulting linear segment, a minimum enclosing rectangle (bounding box) is formed, and a distance value is calculated for each contained spatial position. After resolving multiple distance estimates obtained from overlapping hulls, the heatmap is calculated directly by evaluating the distance values with a distance-dependent function.

Curve parameterization. We assume that a curve $C \subset \mathbb{R}^n$ is described by a finite set of data points $\mathbf{D}_i \in C$, $i = 0, 1, \dots, K$ (cf. (1)). For non-trivial curves with $K > 1$, a smooth parametric B-spline is fitted to the original curve signal. The fitting criterion is based on the formulation by Dierckx[53], where the trade-off between closeness and smoothness of fit is controlled with a smoothing factor S. There, the curve approximation $s(\mathbf{D}_i)$ is optimized such that the sum of residual squares between the approximation and prescribed close interpolation $y(\mathbf{D}_i)$ is equal to or smaller than S so that $\sum_{i=0}^{K} (y(\mathbf{D}_i) - s(\mathbf{D}_i))^2 \leq S$. Following Reisch[54], S is automatically derived from the number of data points with $S = |\mathbf{D}_i| - \sqrt{2|\mathbf{D}_i|}$. This allows the approximating spline to be less sensitive to label or sampling noise. For curves with large local variance in curvature, S can be set to 0, returning an interpolating curve that picks up high frequencies more accurately. Every data point $\mathbf{D}_i$ is assigned a parameter value $t_i \in [0, 1]$. Evaluating t_i over the interval of definition allows to smoothly trace the curve in space[55]. Since we do not make any assumptions on the curve type and its potential physical constraints but still want to keep the closeness of fit when tracing high-curvature regions (especially when the data points are not linearly distributed), we re-parameterize t_i according to the *arc-length parameterization* method[55,56]. To this end, the arc-length parameters are approximated by dividing the domain of the interpolating curve using the chord-length ratios between $\mathbf{D}_i$ and $\mathbf{D}_{i-1}$[56–59].
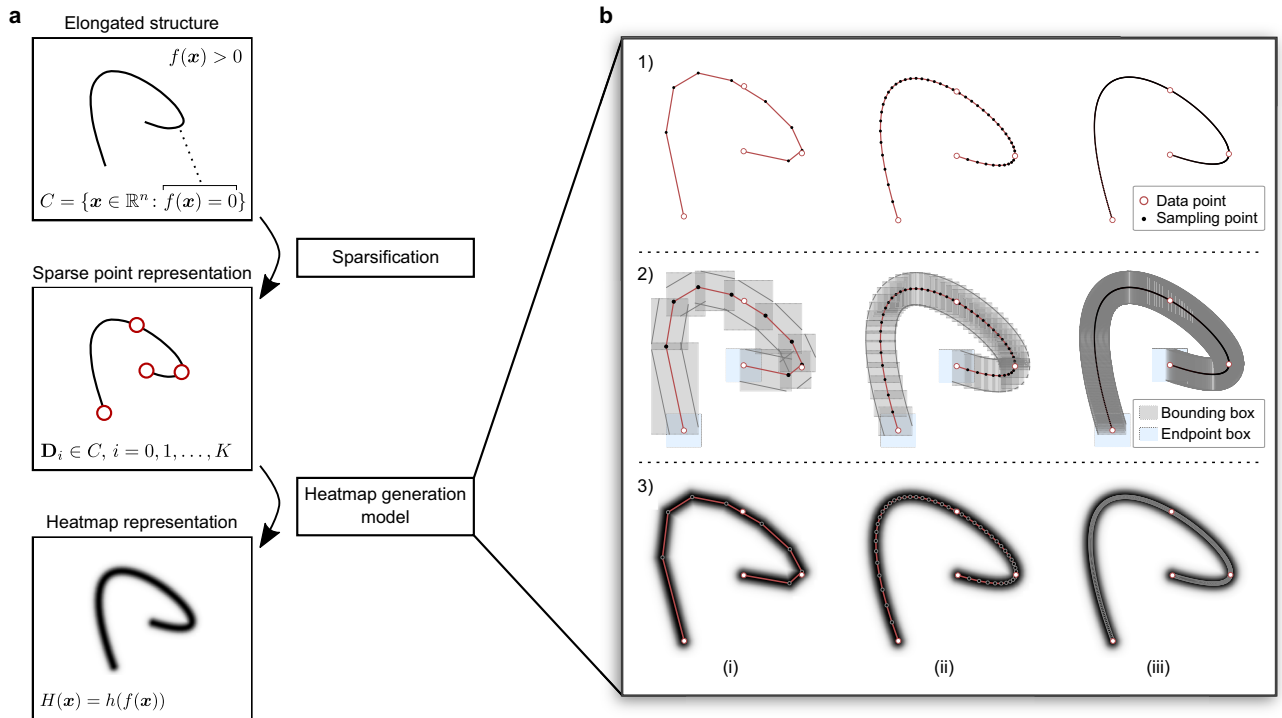
4

Figure 2. (**a**) General workflow of representing an elongated structure as a spatial heatmap. (**b**) heatmap generation model where (i)–(iii) mark 10, 50, and 500 curve sampling points, respectively. (**b1**) Calculation of 2D polylines which connect several sampling points from a parametric interpolation of the original curve signal. (**b2**) approximation of a compact hull that encloses the relevant spatial regions where point-to-curve distances are calculated. (**b3**) Heatmap representation of the curve using a Gaussian-distributed distance-dependent function.

$$t_0 = 0, \quad t_i = \frac{1}{L_h} \sum_{k=1}^{i} \|\mathbf{D}_k - \mathbf{D}_{k-1}\|, \quad t_n = 1 \tag{5}$$

L_h is the length of the polyline connecting the data points and corresponds to the sum of all chord lengths, i.e. $L = \sum_{i=1}^{K} \|\mathbf{D}_i - \mathbf{D}_{i-1}\|$. Floater et al.[58] have shown that this approximation $L_h \approx L$ of the actual curve length L is of second-order accuracy. The coordinates x_i and y_i of the target B-spline control points $\mathbf{P}_i \in \mathbb{R}^n$, $i = 0, 1, ..., K$ can then be determined from the interpolation condition of the parametric spline

$$\mathbf{D}_i = \mathbf{C}(t_i) = \sum_{k=0}^{K} c_k B_{k,d}(t_i) \mathbf{P}_k, \tag{6}$$

where $B_{k,d}(t_i)$ are B-spline basis functions of degree d, and c_k are the respective coefficients. This curve equation can be efficiently solved using Dierckx's algorithm[53]. For $K > 2$ control points, a cubic spline interpolation with $d = 3$ is used, for $K = 2$ the spline is of order $d = 2$, and for a line segment with $K = 1$ it simplifies to a linear interpolation, i.e. $d = 1$. Once the interpolating function is found, m points are sampled from the spline using a uniform sampling of t_i at constant arc-length interval[55]. The connection of these curve points then yields a polyline of $m - 1$ segments. Consequently, the number of segments is an adjustable parameter that regulates the quality of the approximation.

Local distance estimation. In the context of the proposed simplification, the distance evaluation for a single spatial position does not correlate with that of other points in $\mathbb{R}^n$. This enables a piecewise and localized evaluation of the heatmap for each polyline segment.

Following this idea, consider a narrow curve signal on a large background. This background, which can be noise or could provide any other information, has only a marginal contribution to the spatial configuration of this curve. When evaluating $f(\boldsymbol{x})$ and $h(f(\boldsymbol{x}))$, we are only interested in a small local hull around the curve for which we assume compact support. As we work with a polyline that consists of piecewise linear segments, we can translate the idea of a local hull by constructing the smallest enclosing box for each segment. In other words, given the start and end point of a single polyline segment, two parallel segments are set as margins at a distance of $+ w/2$ and $- w/2$ defined by support w. Then, a minimum bounding box enclosing these two parallel lines is defined. When working with image datasets, bounding boxes are ideally axis-parallel to allow for efficient parallel

computation. Since at both the start and end point of the polyline, the resulting hull does not fully extend into the direction of the curve, two additional boxes of width w are centered on these points.

This approach allows the distance function $f(x)$ to be evaluated only for those points x, which are within one or more bounding boxes and therefore significantly contribute to the heatmap. Within each bounding box, the point-to-line distance is calculated for every contained spatial location x. For a line segment with start point l_1 and end point l_2, this distance equates to $f(x) = \left\| x - \left(l_1 + s \cdot \frac{b}{\|b\|} \right) \right\|$ with $a = x - l_1$, $b = l_2 - l_1$, and $s = \max \left\{ 0, \min \left(\frac{a \cdot b}{\|b\|}, \|b\| \right) \right\}$. Since bounding boxes will overlap for non-zero curvatures $\kappa > 0$ or if the curve is self-intersecting, we will likely obtain multiple distance estimates $\mathbf{f}_x = \left(f_1(x), f_2(x), \ldots, f_n(x) \right) \in \mathbb{R}^2$ for some subset of the hull points. This number of individual estimates, i.e., $\|\mathbf{f}_x\|$, also strongly depends on the number of sample points used to construct the polyline and increases for higher sampling frequencies as the ratio between the support w and the distance between sample points gets larger.

In order to map $\mathbf{f}_x$ to a single distance scalar that can be evaluated with $h(\cdot)$, two options for a reduction function $h^*(\mathbf{f}_x)$ are proposed.

Minimum distance. The most simple reduction follows a *winner-take-all* approach where only the estimator with the smallest point-to-line distance is used for evaluating the distance-dependent function $h(\cdot)$.

$$h^*(\mathbf{f}_x) = h\big(\min\{f_i(x) : i \in \{1, \ldots, n\}\}\big) \tag{7}$$

This calculation can be easily parallelized, allowing the heatmap generation algorithm to be executed at linear time complexity $f \in \mathcal{O}(N)$, depending on the total number of spatial positions evaluated in the bounding boxes N (c.f., Supplementary Material Fig. S8).

Inverse distance weighting. An inverse distance weighting (IDW) following Shepard[60] allows the contribution of all distance estimates. IDW assigns an influence weight to each estimate that directly depends on the point-to-line distance.

$$h^*(\mathbf{f}_x) = \frac{\langle \lambda_x, h(f_i(x)) \rangle}{\|\lambda_x\|_1}, \quad \text{where } \lambda_{x,i} = \frac{1}{f_i(x)^p} \tag{8}$$

The power parameter p controls each estimator's relative strength and is set to $p = 1$ (linear) or $p = 2$ (quadratic).

Selection of the distance-dependent function. In principle, any 1D function can be chosen as $h(\cdot)$. However, since the resulting heatmaps are supposed to approximate the spatial position of a curve with some superimposed noise, the **pdf** of a parametric probability function is a natural choice. In the following, we consider various zero-mean **pdf** variants that are either frequently observed signal/noise shapes (Gaussian, Laplace, cosine) or are versatile in their encoding characteristics w.r.t. learning-based algorithms (triangular, rectangular) (c.f. visual representation in Supplementary Material Fig. S9). We also consider a custom signal shape in the form of a simulated hollow-core catheter profile. All distributions are defined with the assumption of compact support $\pm 3\sigma$.

Gaussian:

$$h_{\text{gau},\sigma}(f(x)) = \eta \, \exp\left(-\frac{|f(x)|^2}{2\sigma^2} \right), \quad \underbrace{\eta = \frac{1}{\sqrt{2\pi}\sigma}}_{\text{norm constant}} \tag{9}$$

1D Laplace/2D Super-Gaussian:

$$h_{\text{slp},\sigma}(f(x)) = \eta \, \exp\left(-\frac{|f(x)|}{\sigma} \right), \quad \eta = \frac{1}{2\sigma} \tag{10}$$

Triangular:

$$h_{\text{tri},\sigma}(f(x)) = \eta \, \max\left(\frac{3\sigma - |f(x)|}{3\sigma}, 0 \right), \quad \eta = \frac{1}{3\sigma} \tag{11}$$

Rectangular:

$$h_{\text{rec},\sigma}(f(x)) = \eta \begin{cases} 1 & |f(x)| \leq 3\sigma \\ 0 & \text{otherwise} \end{cases}, \quad \eta = \frac{1}{6\sigma} \tag{12}$$

Raised Cosine:

$$h_{\cos,\sigma}(f(\boldsymbol{x})) = \eta \begin{cases} \frac{1}{2}\left(1 + \cos\left(\pi\,\frac{|f(\boldsymbol{x})|}{3\sigma}\right)\right) & |f(\boldsymbol{x})| \leq 3\sigma \\ 0 & \text{otherwise} \end{cases}$$
$$\eta = \frac{1}{3\sigma}$$

$$(13)$$

Simulated catheter profile:

$$r_o = 3\sigma,\ r_i = sr_o, \quad s \in [0,1]$$
$$y(r) = \sqrt{\max\{r^2 - |f(\boldsymbol{x})|^2, 0\}}$$
$$h_{\text{cat},\sigma,s}(f(\boldsymbol{x})) = \eta(y(r_o) - y(r_i))$$
$$\eta = \frac{2}{\pi(r_o^2 - r_i^2)}$$

$$(14)$$

r_o and r_i mark the radius of an outer and inner circle corresponding to a catheter's outer and inner wall. By changing the scaling parameter s to shrink or expand the inner circle, different wall thicknesses of the tube can be simulated. In general, when interpreting the heatmap as such a **pdf** (which excludes the catheter signal), it is defined to be non-negative and to have a unit integral over its domain. Since we want to use the heatmap in gradient-based learning algorithms in which normalization of the target image is common, we propose to incorporate this normalization in the heatmap calculation such that $\max_{\boldsymbol{x} \in \mathbb{R}^n} h(\boldsymbol{x}) = 1$. We argue that this allows for a more stable gradient across different heatmap parameterizations and reduces the risk of vanishing gradients due to small loss magnitudes. As a result, the normalization constant in (9)–(13) is set to $\eta = 1$, whereas in (14) it is set to $\eta = \sqrt{r_o^2 - r_i^2}$.

Extension to 3D. The localized calculation of relevant distance estimates is easily extendable to 3D. After finding a three-dimensional function to interpolate between a set of data points and extracting a corresponding polyline, the local hull is approximated by extending the two-dimensional bounding boxes to right rectangular prisms. A single prism is formed by calculating four equidistant line segments that are parallel to the polyline segment. A prism width of w allows to obtain the minimum-area enclosing cuboid of the hull that is axis-aligned w.r.t. a Cartesian coordinate system (Supplementary Material Fig. S10).

Analysis of representation properties

In this section, we analyze the method's general numerical approximation error, its behavior for different shape and width configurations, and the relation between sampling ratio and curvature (c.f. Supplementary Material Sect. S2 for analysis of different signal-to-noise ratios). Furthermore, we compare the method's approximation error with that of several related spatial curve representation and approximation methods.

Approximation error for different curvature, heatmap widths, and reduction functions. As shown in Sect. "Heatmaps for elongated structures", the approximation yields exact values for straight lines. However, this is no longer true for curved lines. In the following, we analyze the error introduced by this approximation with distance-dependent weights. We compare the approximation with a numerically evaluated convolution of the curve with the **pdf** of a normally distributed signal.

For this purpose, we devised an experimental setup with a simulated plane curve. By altering the curve's spatial configuration, we can analyze the approximation error for various curvatures, widths, and reduction functions.

Experimental setup to measure the signal representation error. A plane curve C parameterized by three data points $\{\mathbf{D}_0, \mathbf{D}_1, \mathbf{D}_2\}$ is placed on a background of H:2000 × W:2000 px following a spatial configuration of $\mathbf{D}_0 = (1500, 500)$, $\mathbf{D}_1 = (500, [500 + \delta\,1000])$ and $\mathbf{D}_2 = ([1500 - \delta\,1000], 1500)$ (Supplementary Material Fig. S11). The hyper-parameter δ controls the angle between the two curve sections ($[\mathbf{D}_1, \mathbf{D}_0]$ and $[\mathbf{D}_0, \mathbf{D}_2]$) and allows to successively increase the maximum curvature of the curve observed at $\mathbf{D}_0$. For every configuration $\delta \in \{0, 0.1, 0.2, ..., 1\}$, we compute the absolute error ε that accumulates between the definite surface integral of curve C and its numerical approximation at various parameterizations (c.f. Supplementary Material Fig. S11 and Supplementary Material Table S7):

$$\varepsilon = \int_{-3\sigma}^{3\sigma} \left| \int_{\mathbf{S}_i} H(\boldsymbol{s})\,\mathrm{d}\boldsymbol{s} - \sum_{\boldsymbol{e} \in \mathbf{E}_i} l_e h^*(f(\boldsymbol{e})) \right| \mathrm{d}i.$$

$$(15)$$

$\mathbf{S}_i \subset \mathbb{R}^2$ denotes an infinite set of points at an orthogonal distance $f(x) = |i|$ to C while respecting the orientation indicated by the sign of i. Consequently, $\mathbf{S}_0 = C$ (cf. (1)). $\mathbf{E}_i \subset \mathbf{S}_i$ is a finite subset of points that act as evaluation points for the approximated heatmap. The evaluation points $\mathbf{E}_0$, which are located directly on the curve, are chosen by performing a uniform sampling of the curve parameter t. l_e marks the individual chord-length between consecutive points in $\mathbf{E}_0$. All other evaluation points sets $\mathbf{E}_i$ with $i \neq 0$ are sampled such that they are located on the orthogonal line to the curve intersecting the corresponding point in $\mathbf{E}_0$. By assuming a large enough number of evaluation points, we ignore that the distance between the evaluation points is not equally spaced for $f(x) \neq 0$ in the case of non-trivial curves that do not follow a straight line.
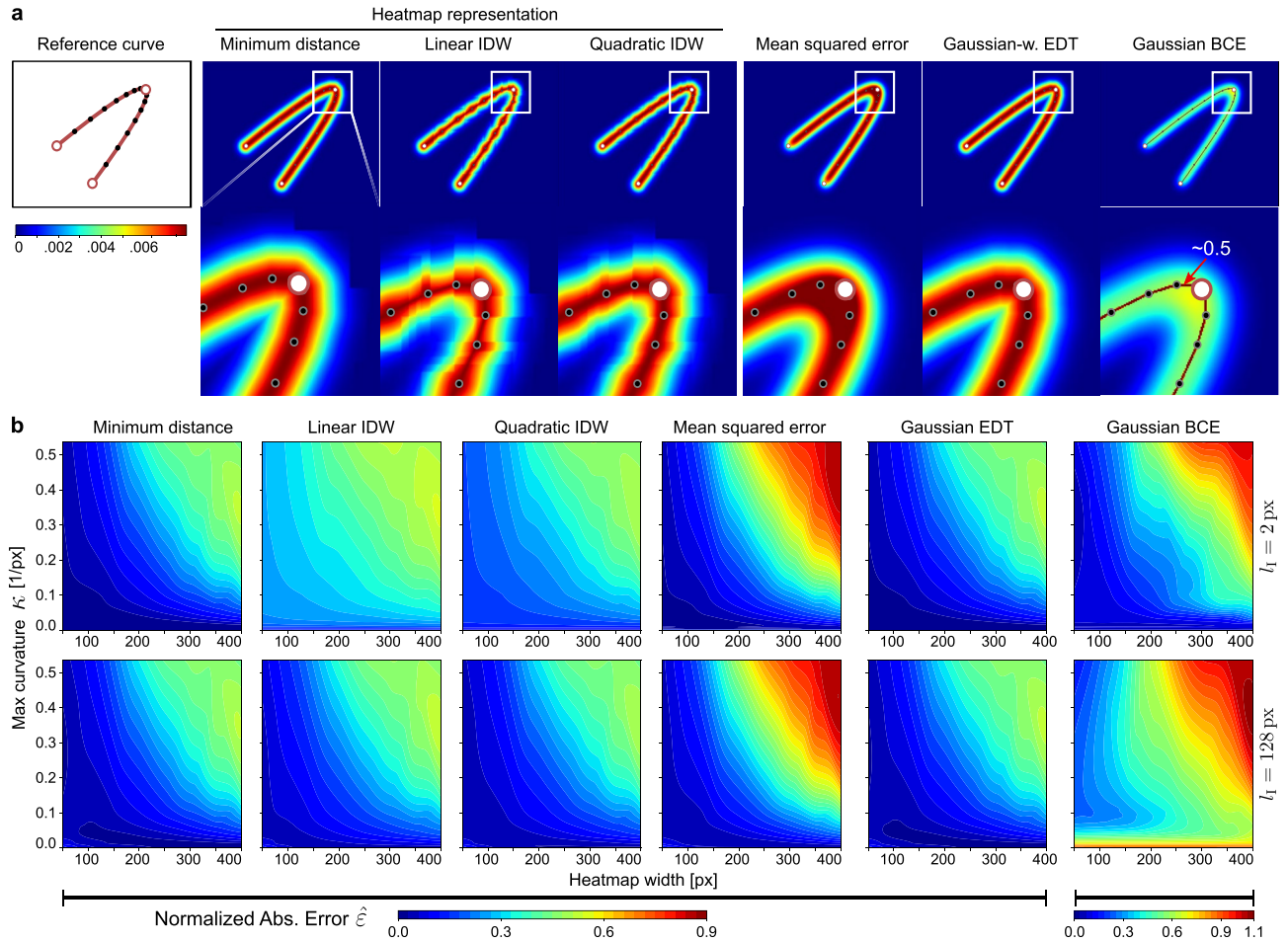
Figure 3. Analysis of approximation errors. (**a**) Comparison of spatial curve representations constructed according to Supplementary Material Table S7 with $\delta = 0.8$, a heatmap width given by $\sigma = 68$ px, and a mean interpolation point distance of $l_I = 128$ px. (**b**) Error contour maps w.r.t. maximum curvature κ and different heatmap widths σ. The color levels encode the normalized absolute error $\hat{\varepsilon}$ between the approximation and the theoretical integral along the curve according to (15).

We can now obtain a normalized error $\hat{\varepsilon} = \frac{1}{L} \varepsilon$ by division with the total curve length L. A normalization across different parameterizations of the support of $h(\cdot)$ is not necessary since we normalize $\int h(f(\boldsymbol{x})) \, d\boldsymbol{x} = 1$ for all σ upon calculating the error values.

The influence of the reduction function $h^*(\cdot)$. In the case of a densely sampled polyline (Fig. 3), both IDW schemes show a substantial noise floor. This can be explained by the presence of many overlapping bounding boxes, which leads to multiple distance estimates for most spatial locations within the hull. This implies an increase in large distance values, which causes a general underestimation of the heatmap values, especially for a linear IDW. Interestingly, this underestimation produces a smooth grid structure for both IDW variants, which particularly emerges in areas of high signal curvature and can be explained by a disproportionate accumulation of large distance estimates originating from oppositely located curve segments. Using the minimum distance circumvents this issue as only the closest estimate is considered, effectively filtering out most estimation noise. Sparse sampling of the curve reduces the error difference between the different reduction functions (Fig. 3), although at the cost of overall precision.

The influence of the curve/heatmap width and curvature. Across all reduction functions, a large curve/heatmap width increases the susceptibility to errors in high curvature sections. This results in error peaks of $\sim 60\%$ of the accumulated signal for maximum curvatures in the range of $\kappa \in [0.25, 0.50] \, [1/\text{px}]$ and widths above ($w \geq 200 \, \text{px}/\sigma \geq 34 \, \text{px}$), which corresponds to 7% and 1.2% of the image diagonal, respectively. Such a value combination implies that the curve at least partly intersects with itself, which is neither considered in the three reduction functions nor the practical implementation (Sect. "Numerical approximation and implementation"). As the heatmap width increases, this overlap area expands, causing a general overestimation of the heatmap values due to high-valued distance estimates (Fig. 4).
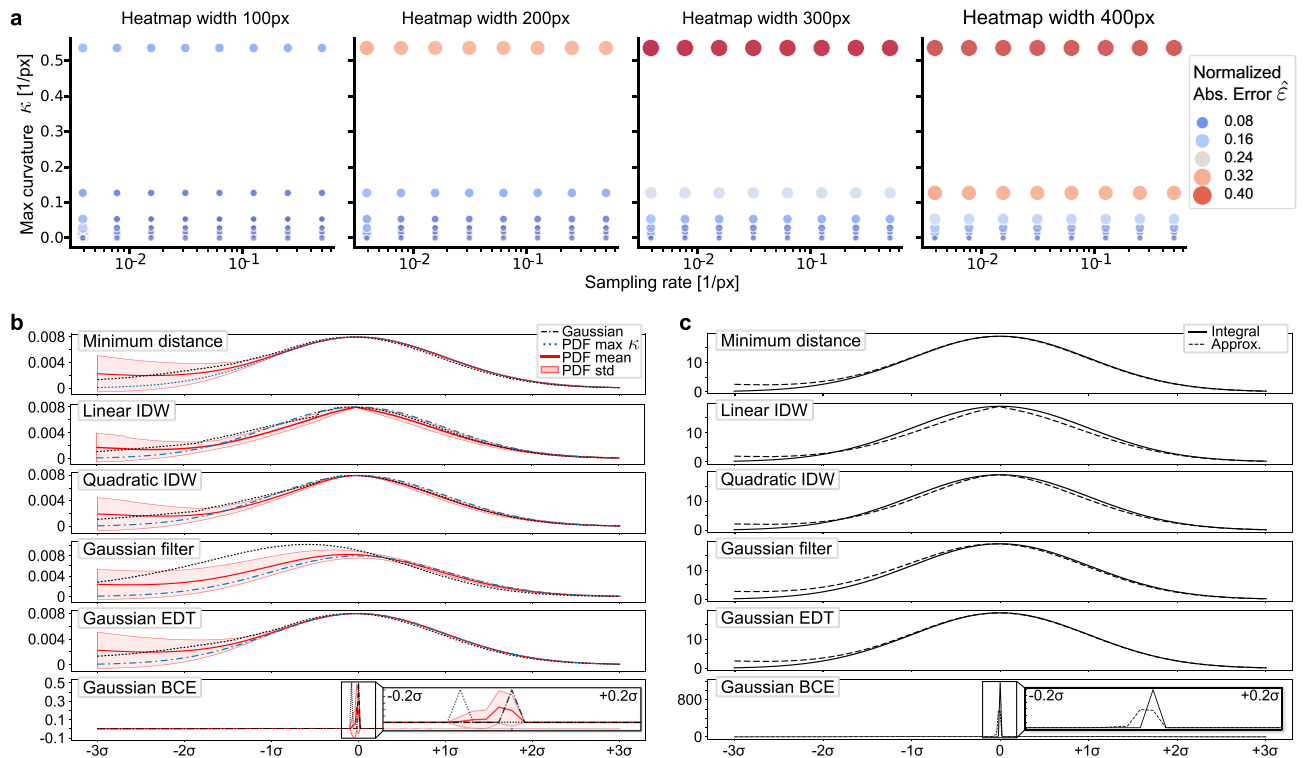
Figure 4. (**a**) Analysis of the relationship between maximum curvature (κ) and sampling rate ($1/l_{\mathrm{I}}$). The size and color of the measurements mark the normalized error $\hat{\varepsilon}$ of the curve's heatmap representation according to (15). (**b**) Curve profiles of the optimal Gaussian **pdf** and the profiles for max curvature and mean/std statistics. (**c**) Accumulation of the approximate and the theoretical integral along the curve ($l_{\mathrm{I}} = 128\,\mathrm{px}, \delta = 0.6$).

The relation of sampling rate and curvature. To relate curvature and sampling rate using the Shannon-Nyquist theorem, we consider their relationship in the context of a curve's highest frequency component. When a curve has high curvature, it undergoes rapid changes in direction, which corresponds to higher frequency components in the curve. The maximum curvature (κ) is given as the reciprocal of the radius of curvature and represents the curve's maximum bending. The maximum curvature can be related to the highest frequency component through $f_{max} = \kappa/(2\pi)$. Now, the sampling theorem mandates a sampling rate at least double the highest frequency component to avoid aliasing, leading to the inequality $r > 4\pi\kappa$ for the sampling rate r. This relationship guides the selection of an appropriate sampling rate during sparse sampling of the curve, ensuring accurate capture of the curve's curvature and avoidance of aliasing effects.

At the same time, recall that we explicitly model the spatial uncertainty of the curve's position, resulting in non-zero widths in the heatmap representation. The sampling quality thus also strongly depends on the chosen heatmap width. This is shown in Fig. 4, where the approximation error increases for larger curvatures across all sampling rates (reciprocal of sampling point distance $1/l_{\mathrm{I}}$). Although there are small improvements in approximation quality from higher sampling rates, these must be weighed against the higher computational requirements. Moreover, since the width of the heatmap shows the highest impact on approximation fidelity, we argue that a self-overlapping of the curve or its noise spread is the main reason for larger errors in the approximation. Consequently, high curvatures can be most effectively tackled by choosing a small-to-medium-sized heatmap width (if possible). In the practical setting, signals with maximum curvature of up to $\kappa \approx 0.1[1/\mathrm{px}]$ can be modeled with negligible errors with little requirements w.r.t. the sampling rate.

Comparison to related curve representations. Using the same experimental setup as in Sect. "Approximation error for different curvature, heatmap widths, and reduction functions", we compare the method's error to that of several related spatial curve representations. These share similarities in that they evaluate the point-to-curve distance, use a Gaussian model to assess the positional likelihood of the curve points or are rooted in a probabilistic interpretation of conventional cost functions in optimization algorithms (Fig. 3).

Mean squared error on curve points. We construct a spatial representation of the curve by filtering the discretized curve points with a Gaussian low-pass filter. This rationale can be interpreted as an evaluation of the mean squared error of the individual curve points, where the squares of the errors correspond to the exponent in the **pdf** of a Gaussian distribution and a constant normalization offset. The approximation errors obtained by this representation under various configurations are illustrated in Fig. 3. Since the number of convolution operations per spatial position increases sharply in the near-curve region with high curvature, we see high approximation errors with increasing signal complexity. This aliasing leads to a systematic broadening and migration of the

9

distribution peak towards the inside of the curve, which reduces the likelihood of the curve points given the Gaussian model (Fig. 4).

Gaussian-weighted Euclidean distance transform. Discretizing the curve points lets us derive a representation of the curve as a distance field by means of Euclidean distance transform (EDT)[61]. Each position in this field corresponds to the Euclidean distance to the nearest discrete curve point (Fig. 3). Following (4), these distances can be elegantly evaluated with a Gaussian **pdf**. The observed approximation errors are almost identical to the proposed numerical approximation, and the smoothness of the curve similarly increases with the number of sampling points. However, for configurations with lower spatial resolution and more compact support of the **pdf**, the curve discretization and the resulting binning of distance estimates introduce stair-step artifacts (Supplementary Material Fig. S8) that could be detrimental in some scenarios. However, for extremely large noise spread and very dense curve sampling, this representation might be preferred due to very efficient computation with linear time complexity $f \in \mathcal{O}(N)$[61] w.r.t. the number of image pixels N with no dependency to the number of sampling points and σ. In such a scenario, no time can be saved by evaluating only the numerically relevant distances in the regions near the curves (as in the proposed representation) due to excessive bounding box overlap (Supplementary Material Fig. S8).

Gaussian-weighted BCE. We implement the rationale of distance weighting directly in a BCE loss function, which is typically used for optimizing binary segmentation problems. For that purpose, a discrete 2D convolution of the curve points and the impulse response of the BCE term is calculated. Each position in the impulse response is weighted using a Gaussian **pdf** (Fig. 3). The approximation error is calculated by comparing the resulting curve profile with the corresponding 1D impulse response. Similar to the MSE variant, we observe peak flattening and migration towards the inside of the curve, which is increased in high curvature areas with large noise spread. Interestingly, sparse sampling of the interpolation function results in larger errors in regions with little curvature. Due to the impulse response's extremely narrow and high-magnitude peak, even minor offsets between the original curve and the approximation lead to large errors. This behavior is a direct result of the B-spline interpolation scheme, where the relative amount of low-curvature sections in the curve legs increases with a higher curvature at the turning point.

CNN-based estimation error for different signal and heatmap configurations.

A primary application of the presented representation is its use as an operational description of the target function in a CNN-based learning system. This raises the question of how the configuration of the observed type of target signal that should be approximated responds to different the configuration of the heatmap. To address this question, we use a signal simulation model to construct pairings of varying signal and heatmap configurations and compare the predictive performance of a reference CNN (Sect. "Simulated signals on chest X-ray", Supplementary Material Table S8) using the Average Symmetric Surface Distance (ASSD) metric (Sect. "Variant 1: ASSD"). Here, we take advantage of the fact that our method not only allows us to construct heatmaps that approximate a specific signal but also enables the simulation of an arbitrary elongated structure itself. To generate non-trivial and realistic input images for the neural network, the simulated structures are superimposed on the human chest radiographs (Sect. "Simulated signals on chest X-ray").

The relation between a signal distribution and heatmap distance-dependent function. We observe a weak correlation between the signal distribution and the distance-dependent function of the heatmap (Fig. 5). Only using a rectangular distribution leads to a substantially worse reconstruction of the original curve, with the ASSD metric reaching error values of up to 5 px. Mesokurtic choices of the distance-dependent function (Gaussian, Raised Cosine) generally work well for all types of signal configurations. The custom catheter signal (Sect. "Selection of the distance-dependent function") is estimated with negligible error for all heatmap configurations. Similar to the rectangular distributed signal, the signal strength plays a minor role due to high intensities across the discrete support.

The influence of signal and heatmap width. A perfect match between the signal and heatmap width does not benefit the representation quality during network inference. On the contrary, a small heatmap width is generally favorable regardless of the signal configuration. This observation is of great advantage in real-life applications where we cannot assume a fixed signal width or where little information of the signal configuration is available. As shown in Fig. 5, the rectangular distribution shows the largest error with increasing heatmap widths. Due to its uniform intensity profile, a high width forces the network to significantly overestimate the signal intensity distribution, especially for leptokurtic signals (Laplace, Triangular). This representation bias is slightly reduced upon the optimization with BCE but does not change the general error tendency.

The limitations by signal strength. For all but a rectangular signal shape, the strength of the signal overlay and, consequently, the separability between background and signal based on image information plays a dominant role. For weaker signals with $a = 0.15$, the predicted heatmaps suffer from increased fragmentation, frayed edges, and underestimated intensity values. Especially for wide and leptokurtic signals, such little contrast significantly impedes the reconstruction quality of the original signal curve. This behavior is expected since small deviations of the center-line during network inference can lead to much stronger mass shifts than what can be observed for mesokurtic or platokurtic signals, directly affecting the determined Otsu threshold and skeletonization.
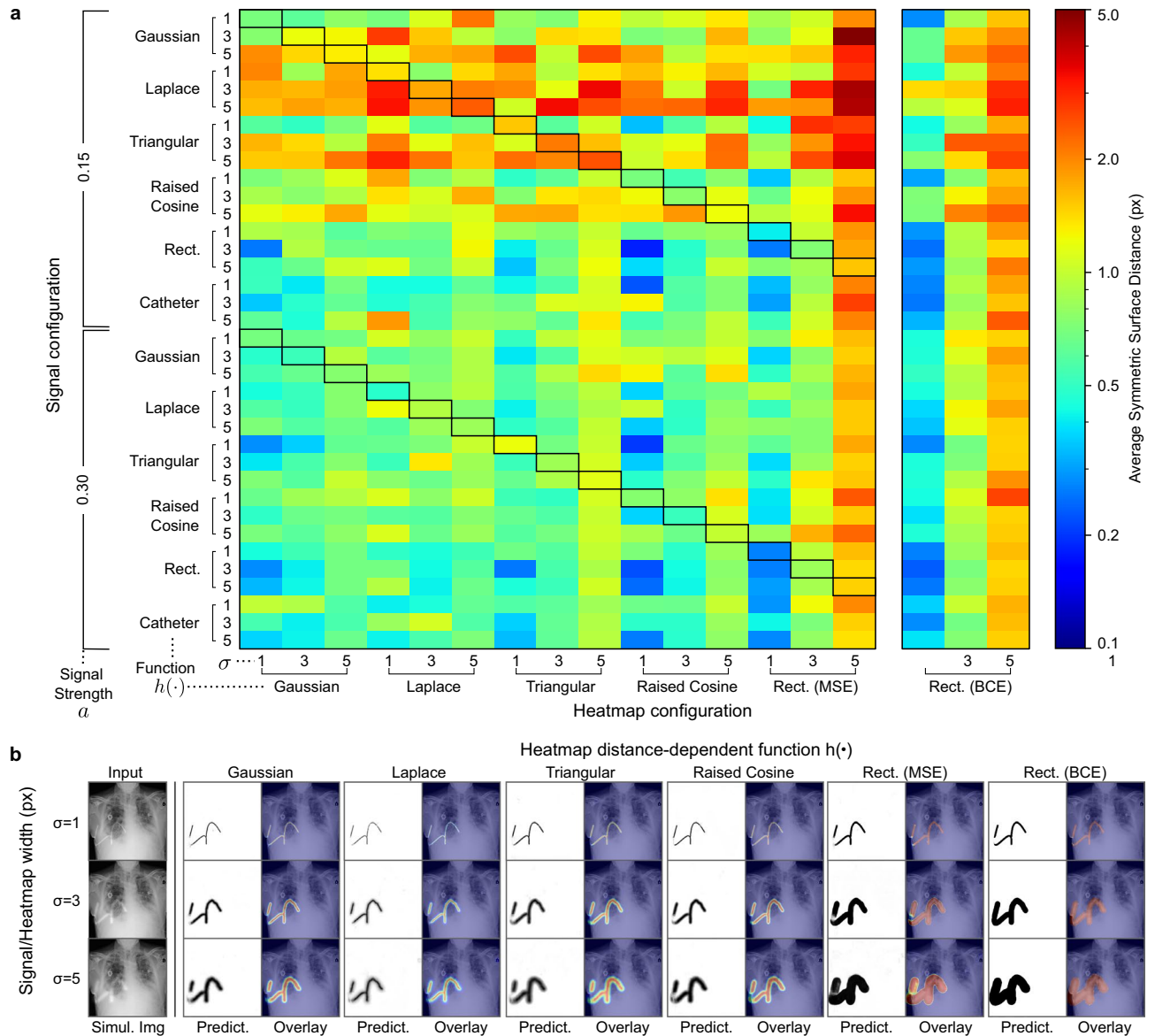
Figure 5. CNN estimation error for various signal/heatmap configurations. (**a**) Heatmap of mean test ASSD for signal/heatmap configurations. a: signal overlay strength, $\sigma_{\text{sim}}/\sigma_{\text{hm}}$: signal/heatmap width as standard deviation (px), $h(\cdot)$: distance-dependent function. Black boxes mark matching signal and heatmap configurations. (**b**) CNN predictions for Gaussian-distributed signals. The test sample was selected according to the smallest difference to the mean ASSD over all signals with $a = 0.15$.

Application to 2D and 3D representation problems

In the following section, we evaluate the applicability of our approach to real-life application examples. Complementing evaluation on other datasets and curve types is provided in Supplementary Material Sects. S3–S6.

Anatomical structure detection on knee radiographs. We evaluate our method for the detection of anatomical structures and a surgical reference line on 2D radiographs (Sect. "Datasets and training protocols"). We analyze heatmap representations of four elongated anatomical features that are part of the human knee joint, ranging from edges/boundaries, curved and straight lines partially supported by image information, and a contextual center-line[62,63] (Supplementary Material Fig. S12): (A) Blumensaat line in the intercondylar notch of the distal femur (curved). (B) Mean contour of the medial and lateral femoral condyles (curved). (C) Plateau line of the proximal tibia (straight). (D) Anatomical axis of the femur bone (straight).

Features (A)-(C) are structures that can be *directly* delineated on lateral radiographs by perceivable contrast in the respective areas or by semantically connecting salient landmarks. The anatomical axis (D) is not directly derived from image information but instead relies on the global orientation of the bone given by the surrounding bone shaft contours. We are particularly interested in the learning algorithm's capability to infer such *contextual* structures. The directly perceivable features were trained jointly using a single model, and the contextual axis feature was trained individually.
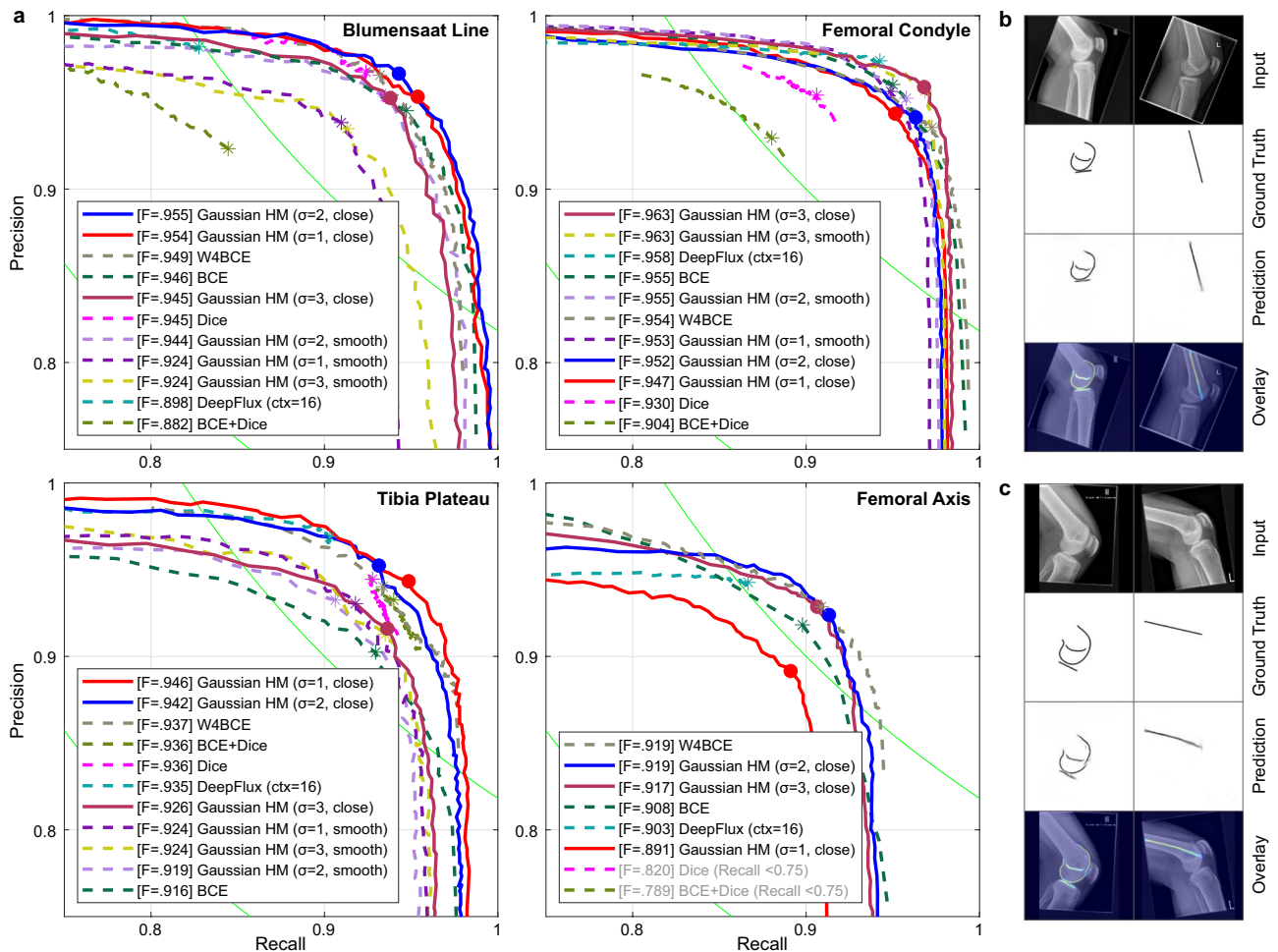
Figure 6. Performance analysis for representation of diverse anatomical features. (**a**) PR curves with F-measure (ODS). (**b**) Low-error predictions. (**c**) Erroneous samples.

As shown in Fig. 6, Table 1, and Supplementary Material Fig. S13, the heatmap predictions for both *direct* and *contextual* anatomical structures are of overall high spatial precision. The F-measure for the contextual femoral axis (D) is slightly lower than the corresponding scores for direct features (A)-(C) which are supported by salient anatomy. The observed outliers with low individual F-measure scores are caused by anatomical deviations, slight misalignment of the femoral condyles, or joint deformities such as ventral/dorsal bending of the bone. For this task, we can derive a relation between the optimal heatmap width σ_{HM} and the type of structure to be represented. While we observe better detection rates for delicate structures like the Blumensaat line and plateau line for smaller widths, more pronounced and larger structures like the femoral condyle and axis benefit from a larger width. Furthermore, a close interpolation is preferable, which is particularly evident for intricate structures with detailed curves like the Blumensaat line. This difference, however, is negligible for larger structures like the femoral condyle. Interestingly, although the tibial plateau is a straight line and its interpolation doesn't change upon different smoothing conditions, its detection performance benefits from using a close interpolation for the other two features in the combined training. In comparison, the segmentation and skeletonization methods achieve high but inferior F-measure scores compared to the heatmap representation. In particular, optimization with a Dice loss or the combination of Dice and BCE is significantly worse, achieving comparatively low recall for the femur axis.

Detection of surgical implants in 3D CBCT volumes.

Surgical treatment of fractures or joint deformities is often guided by 3D cone-beam computed tomography (CBCT). Within these volumes, knowing the position and orientation of metallic implants like screws and k-wires is crucial to ensure the effectiveness and safety of the intervention. Using this task as an example, we investigate the use of 3D heatmaps and their applicability for representing elongated 3D structures (Sect. "Datasets and training protocols"). For evaluation and reporting, we use previously described signal extraction and ASSD metric (Sect. "Signal extraction and evaluation metric"), which can be easily extended for 3D curves.

Both screws and k-wires are reliably detected with high spatial precision (Fig. 7). Even in complex cases where a substantial number of implants are close to each other, the orientation and shape of each object instance are inferred with reasonable accuracy. However, in direct comparison to the 2D examples, the prediction quality is

Method	Blumensaat line	Femoral condyle	Tibia plateau	Femur axis
BCE	0.946	0.956	0.916	0.908
Dice	0.945	0.930	0.936	0.820
BCE+Dice[1]	0.882	0.904	0.936	0.789
W4BCE[2]	0.949	0.954	0.937	**0.919**
DeepFlux-P[3]	0.898	0.958	0.935	0.903
Gaussian heatmap				
$\sigma_{\mathrm{HM}} = 1$ (smooth)	0.924	0.953	0.924	N/A
$\sigma_{\mathrm{HM}} = 2$ (smooth)	0.944	0.955	0.919	N/A
$\sigma_{\mathrm{HM}} = 3$ (smooth)	0.924	**0.963**	0.924	N/A
$\sigma_{\mathrm{HM}} = 1$ (close)	0.954	0.947	**0.946**	0.891
$\sigma_{\mathrm{HM}} = 2$ (close)	**0.955**	0.952	0.942	**0.919**
$\sigma_{\mathrm{HM}} = 3$ (close)	0.945	**0.963**	0.926	0.917

Table 1. F-measure (ODS) comparison of different methods to detect anatomical structures on knee joint radiographs. Corresponding analysis with the ASSD metric is provided in Supplementary Material Table S9. Comparison methods are (1) BCE+Dice[27], (2) BCE + Dice[26], (3) DeepFlux-P[35] (ctx = 16). Best value for each anatomical structure is in bold.

slightly lower. For all test data with metal artifact reduction (MAR) (Sect. "Datasets and training protocols") and a Gaussian reconstruction filter size of $\sigma_{\mathrm{rec}} = 1$, a median ASSD of 4.59 voxels, $CI_{95\%}[3.65, 7.43]$ is achieved. Qualitative analysis of the high-error cases reveals that false-positive response at metallic plates, a heatmap intersection of close neighboring objects, and over-/underestimation of object length are the main reasons for failure. The latter can be attributed partly to ambiguous ground truth in anatomically complex and cluttered regions. Interestingly, using the same filter size but no MAR method yields a comparable score of 5.36 voxels, $CI_{95\%}[3.79, 6.66]$. This suggests that the proposed 3D heatmap representation can successfully overcome a reduction in image quality and that meaningful heatmap extrapolation in compromised areas is possible.

Discussion and conclusion

This work introduces and investigates a general and versatile mathematical description for curves in space of various origins, shapes, and information levels. The primary purpose of this representation is to be used as a target in optimization-based algorithms that should learn an encoding of the structures and approximate their location in space. Given the distribution of a curve in space and an estimate of its positional uncertainty, a heatmap is generated by convolving this distribution with a filter function. Instead of a time-consuming numerical integration, we approximate the convolution by evaluating a freely selectable distance-dependent function. Since this function depends exclusively on a single spatial position, this simplification allows a localized, piece-wise, and thus easily parallelizable evaluation of the heatmap-generating function. This not only reduces the wall clock time in practical applications but also promotes a lightweight technical implementation.

Analysis of the representation properties with synthetic and real-life application examples allows several interesting conclusions: (1) The proposed heatmap representation can approximate arbitrary curves/signals as long as no extreme curvatures (e.g., close to a directional change of 180°) are observed. (2) Calculating the heatmap with a mesokurtic function and small or medium overall width is preferred for most signal/curve types. Especially for filigree structures, it is advantageous to assume a low position uncertainty of the curve in space. (3) Even in the case of degraded image quality caused, for example, by shadowing, occlusion, or low SNR, the estimation of the heatmap by a CNN is robust and sufficiently accurate. Depending on the severity of the degradation, missing image information can be fully or partially extrapolated. (4) Both structures that are displayed directly in the image as well as structures derived from context can be approximated well. (5) The application to a 3D detection task shows encouraging results, although at a higher overall error when compared to heatmaps in 2D.

Comparing the heatmap representation to previous techniques allows us to derive advantages, drawbacks, and implications for practical applications. While similarly expressive and flexible as (binary) segmentation techniques, heatmaps inherently allow to incorporate prior knowledge of the expected signal distribution and its positional noise. In practice, this can be valuable for signals that expose some inherent structure of their orthogonal profile, e.g., a hollow-core metal rod observed via transmissive imaging. Moreover, we can more precisely represent structures with known physical parameters without having to perform explicit physical modeling, e.g., using the degree of stiffness/flexibility to determine plausible deviations of the nominal curve position. In contrast to related approaches, the proposed representation model also allows for instance-level parameterization of multiple individual structures within the same output channel. Furthermore, data augmentation can be applied to the numerical sparse point representation, eliminating interpolation noise observed in spatial transformations of the optimization target, e.g., a segmentation mask. An inherent downside of the approach is that the visual appearance of a signal with varying width (e.g., naturally non-uniform signals, perspective changes due to out-of-plane components, etc.) cannot be accurately modeled. While the learning algorithm can, to some degree, generalize for these cases, purely spatial methods like segmentation are much more suited for such tasks.
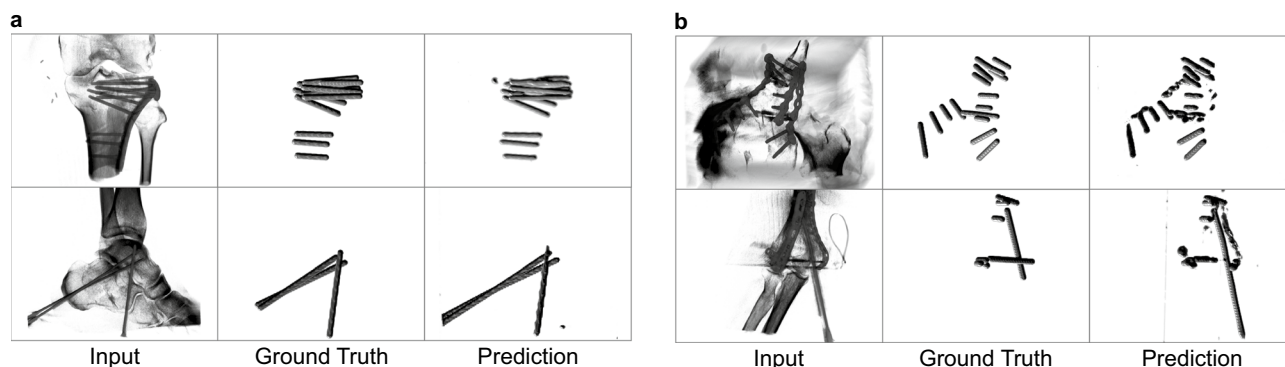
Figure 7. Analysis of detection performance of surgical screws and k-wires on 3D CBCT volumes. (**a**) Low-error predictions. (**b**) Erroneous samples.

Furthermore, for a dense accumulation of elongated structures with potential overlap, the method becomes much less efficient than a single-shot global representation like most segmentation methods.

In addition to these drawbacks, we want to note that no analysis has been performed to date on a suitable method that yields a parametric reconstruction of the original signal from the heatmap representation. Although a thinning approach yields satisfactory results in the pixel/voxel space, obtaining a functional description is not trivial, especially if the heatmap is fragmented or self-intersecting. This problem further intensifies if several heatmaps are close to each other and are approximated in the same output channel of a neural network. Consequently, separating structures into separate output channels would be desirable, even if no matching of object instances between multiple input images/volumes is possible. Also, using a neural network to directly predict the knots and parameter values of B-spline curve approximations[64] is an attractive direction for future extensions of this work.

Methods and materials

Signal extraction and evaluation metric.
Two post-processing protocols are used for signal recovery and empirical evaluation, depending on the curve type and the respective dataset: (1) Center-line extraction by medial axis transformation and subsequent comparison of the ASSD between ground truth and prediction curves. (2) Non-maximum suppression of the heatmap, morphological thinning, and subsequent calculation of the F-measure between ground truth and prediction curves.

Variant 1: ASSD. A signal estimate C_{pred} is recovered as a set of points that satisfy the equality in (1) using the subsequent post-processing scheme: 1) Thresholding of the estimate with Otsu's method[65] to obtain a binary image. (2) Medial axis transformation to obtain a curve representation of unitary width. To this end, we use a thinning of the binary estimate with the skeletonization method by Zhang et al.[66].

To estimate the quality of fit of such estimate, we use the ASSD metric[67]. A distance measure d is defined as the minimum Euclidean distance of a point $\boldsymbol{x}$ on a curve C to all points $\boldsymbol{x}'$ on a second curve C', which is computed by $d(\boldsymbol{x}, C') = \min_{\boldsymbol{x}' \in C'} \|\boldsymbol{x} - \boldsymbol{x}'\|_2$. The ASSD between the two curves C and C' is then calculated by evaluating this distance for every point $\boldsymbol{x} \in C$ and $\boldsymbol{x}' \in C'$ and averaging the resulting distance values

$$\text{ASSD} = \frac{1}{|C| + |C'|} \left(\sum_{\boldsymbol{x} \in C} d(\boldsymbol{x}, C') + \sum_{\boldsymbol{x}' \in C'} d(\boldsymbol{x}', C) \right)$$

In theory, this evaluation policy has the advantage of being invariant to the width and distribution of the simulated signal and estimated heatmap. However, it has to be noted that the employed thinning can cause branching artifacts in case of noisy and self-intersecting heatmap estimates. Using a larger signal or heatmap width also implicates a higher upper limit of the possible length of such subsidiary branches. However, a qualitative review of the high-error cases revealed that this effect is negligible as most errors are caused by signal fragmentation in ambiguous regions.

Variant 2: F-measure. To conform to common evaluation protocols in boundary detection and skeletonization, we interpret the heatmap as a noisy curve probability map. First, a standard non-maximum suppression (NMS) is used to suppress curve point estimates for which a relatively stronger estimate is observed in orthogonal direction within some radius r_{spr}[68]. Second, the probabilities are converted to binary curve labels using some probability threshold. Third, iterative morphological thinning is employed until an edge of unitary width is obtained. The curve estimate is then compared to the ground truth curve by calculating the minimum-cost correspondence (i.e., Euclidean distance), evaluating it against a maximum distance tolerance d_{max}, and computing the F-measure $F = 2\,PR/(P + R)$, where P and R mark the Precision and Recall, respectively[46]. Final reporting is done using the best curve selection threshold across all images in the dataset, i.e., at optimal dataset scale (ODS) or at optimal image scale (OIS)[69]. Individual configurations of r_{spr} and d_{max} are provided in Sect. "Datasets and training protocols".

Neural network architecture. For the 2D experiments, we use a single Hourglass Module which is a standard model for heatmap-based landmark detection[70]. The model is configured with a feature root of 128 used across all abstraction levels, ReLU activation functions in the bottleneck blocks, and instance normalization layers to obtain a smoothed optimization landscape in the presence of different contrast levels[71]. The bottleneck blocks are constructed in the standard 3-layer structure with half the number of convolution kernels in the actual bottleneck mapping. To facilitate true identity mapping without applied non-linearity, we use a pre-activation layout according to He et al.[72]. The 3D architecture is specified in Sect. "Surgical implants in CBCT volumes".

The network parameters are initialized with the He initialization strategy[73]. The cost function operates on the output of the last layer and performs a heatmap-matching between network prediction and generated ground truth data. The default choice for the cost term is MSE, which is replaced by BCE in the segmentation experiments in Sect. "CNN-based estimation error for different signal and heatmap configurations". Implementation was done in PyTorch v1.6.0[74] (Python v3.8.5, CUDA v11.0, cuDNN v7.6.5). All experiments were performed on an x64 machine with an NVIDIA Titan RTX GPU and 64 GB memory. Details on data augmentation, data input standardization, and the remaining task-specific hyper-parameters are provided in the dataset descriptions in Sect. "Datasets and training protocols".

Datasets and training protocols. *Simulated signals on chest X-ray.* We analyze the proposed representation's characteristics for different signal and heatmap configurations using simulated signals superimposed on chest X-ray images. First, a simple heuristic generation model is devised for the simulation of realistic curve signals. Therefore, a finite number of data points $\mathbf{D}_i$ with $|\mathbf{D}_i| \sim \mathcal{U}\,[3, 10)$ is spatially distributed on a zero-valued envelope of size H:256 × W:256 px. The location of the seed point $\mathbf{D}_0$ is randomly initialized such that it lies within a slightly more narrow sub-region to avoid too short and truncated curves: $\mathbf{D}_{0;x}, \mathbf{D}_{0;y} \overset{i.i.d.}{\sim} \mathcal{U}\,[10, 246]$. Based on this seed point, the x, y coordinates of the subsequent points are calculated as $\mathbf{D}_{i+1;x} = \mathbf{D}_{i;x} + s \cos \phi$ and $\mathbf{D}_{i+1;y} = \mathbf{D}_{i;y} + s \sin \phi$ with $s \sim \mathcal{U}\,[20, 60)$ and $\phi \sim \mathcal{U}\,[-90, 90)$. If a newly sampled point exits the sub-region, a new seed point is generated, effectively sampling a new data point distribution. The parameters of the simulation model were chosen heuristically such that realistic global linear and angular moments are preserved over the point distribution. This promotes a minimum amount of stiffness and directional flow of the parameterized curve, which can be assumed for most real-life elongated structures such as surgical catheters or trivial skeletons. Upon the optimization of neural network parameters, the signal data points are simulated on the fly to maximize the number of curve characteristics and aid subsequent generalization. This online simulation is subject to a fixed global random seed to ensure comparability between different configurations. The data points that are used for validation are predefined and not changed between epochs or configurations.

Using the signal simulation model, we calculate a heatmap representation $H_{\text{sim}}(\boldsymbol{x})$ of the signal to be superimposed on a realistic background image X_{img} (configuration provided in Supplementary Material Table S8). Upon fusion of signal and image, the heatmap is multiplied by a strength factor a, and an i.i.d. sampled additive Gaussian noise Z is added. We obtain the final simulated image as $X_{\text{sim}} = X_{\text{img}} + (a \cdot H_{\text{sim}}(\boldsymbol{x})(1 + Z))$ with $Z \sim \mathcal{N}(0, \sigma_Z^2)$. The background images stem from publicly available chest X-ray data[75] originating from the *COVID-19 image data collection*[76] and *ChestX-ray8*[77] datasets. The 1125 images were partitioned into two cohorts for training and validation following a 900/225 (80%/20%) split. For comparison and reporting, we evaluated the metrics using the best model parameters for each signal/heatmap configuration directly on the validation dataset using the ASSD metric (Sect. "Variant 1: ASSD").

Optimization is done with an Hourglass model for 40 epochs, a batch size of 2, a learning rate of 0.00025, and L2 regularization with a factor of 0.00005 using the RMSProp update policy. The number of interpolation points for polyline sampling is set to 200. An online augmentation sequence is applied to the training images, which involves minor shearing/axes shift i.i.d. sampled from $[-5, 5]$ px, rotation in the range of $[-5, 5]°$, image scaling by a factor of $[0.9, 1.1]$ %, and a center crop which truncates $[0.0, 0.1]$ % of the height or width for each image side independently. Before training and inference, each image is brought to a spatial resolution of H:256 × W:256 px by center-cropping the larger dimension w.r.t. the target aspect ratio followed by a cubic down-sampling.

Anatomical structures on knee radiographs. Network optimization and evaluation are performed on 223 clinical X-ray images retrospectively collected from anonymized databases. The images stem from different patients, are centered on the knee joint, and were acquired as strictly lateral projections with both distal femoral condyles superimposed with closely aligned contours. A medically trained engineer annotated the ground truth line segments corresponding to each anatomical structure with the labelme annotation tool[78]. A representative subset of 38 ground truth annotations was reviewed and verified by three expert trauma surgeons. The resulting dataset was split into two cohorts of 174/49 (80%/20%) for training and validation with disjoint patients. For the method comparison, we evaluate the metrics using the best model state on the validation dataset without separate analysis on a hold-out test dataset. We separate optimization between *direct* and *contextual* structures (Supplementary Material Fig. S12) using two separate Hourglass networks (Sect. "Neural network architecture"). The training was done for 250 epochs with a batch size of 2, a learning rate of 0.00025, and L2 regularization with a factor of 0.00005 using the RMSProp update policy. As each anatomical structure is always present in the image and has a unique appearance, we do not combine the structures into a single heatmap but rather use three distinct output channels for the direct structures. To aid generalization, the augmentation pipeline described in Sect. "Simulated signals on chest X-ray" is extended by horizontal flipping. A standardized training and evaluation resolution of H:256 × W:256 px is obtained by resizing (retaining the aspect ratio) and a final center-crop. To evaluate potential effects of strong local curvature, e.g., for the Blumensaat line, we compare curve approximation variants with (S > 0; *smooth*) and without (S = 0; *close*) smoothing condition (Sect. "Numerical approximation and implementation"). Method evaluation is done with the ODS F-measure (Sect. "Variant 2:

F-measure"). The distance tolerance for matches between edge predictions and ground truth is set to a standard value of $d_{max} = 0.0075$ w.r.t. the image diagonal[69]. The suppression radius during NMS is set to $r_{spr} = 1$.

Surgical implants in CBCT volumes. The dataset consists of 141 3D CBCT volumes recorded across a variety of different body regions: acetabulum, calcaneus, cervical/thoracal/lumbar spine, humerus, distal tibia, proximal tibia, and wrist. Each volume was acquired with a spatial resolution of L:512 × H:512 × W:512 voxels of size 0.31 mm and was down-sampled to L:128 × H:128 × W:128 voxels to fulfill the constraints of available hardware. Each volume contains between 1 and 16 metallic screws or k-wires inserted into the bone for fragment fixation, ligament reconstruction, or plate osteosynthesis. The corresponding 3D ground truth volumes were generated according to the 3D extension explained in Sect. "Numerical approximation and implementation" and are based on two control points obtained by manual annotation of both the head and tip of each screw/wire. The data is split into training and validation cohorts of 101 and 40 volumes, respectively. Training and validation is done for six different reconstructions of each volume, differing in the size of the Gaussian reconstruction filter ($\sigma_{rec} \in [1, 2, 4]$) and in the application of a metal artifact reduction (MAR) technique (frequency-split MAR (FSMAR)[79] and no MAR). Optimization is performed for 50 epochs using a 3D U-Net architecture[80] with residual connections within each encoding and decoding block. The feature root is set to 16 and is doubled at each of the five encoder levels. We use a batch size of 1 and a learning rate of 0.005. Since each patient's implant configuration differs, a single output heatmap volume is optimized. The ASSD metric (Sect. "Variant 1: ASSD") is used to evaluate the quality of the reconstructed centerline of the screw/wire.

Ethics approval. The proprietary clinical data was obtained retrospectively from anonymized databases and was not generated intentionally for the study. For this study type, formal consent is not required. All other data used within this study originates from publicly available datasets.

Data availability

The main data supporting the results in this study are available within the paper and its Supplementary Information. Primary and secondary datasets are disclosed as follows. (Primary datasets)—the simulated polylines used for analysis on synthetic data are available from the corresponding author, F.K., upon reasonable request. The 3D CBCT volumes and 2D anatomical structures data are not publicly available due to them containing information that could compromise intellectual property of Siemens Healthcare GmbH. (Secondary datasets) – all secondary datasets are publicly available. At the time of publication, the they can be downloaded from official repositories as follows: The Berkeley Segmentation Dataset 500 (BSDS500) and evaluation suite is available at https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/. A pre-processed version of NYU Depth Dataset V2 (NYUD) following the data protocol in Gupta et al.[81] and Xie & Tu[69] is available at http://mftp.mmcheng.net/liuyun/rcf/data/NYUD.tar.gz. Skeletonization data for WH-SYMMAX is available at http://data.kaizhao.net/projects/skeleton/wh-symmax.zip, for SK-LARGE at http://data.kaizhao.net/projects/skeleton/sk1491.tar.gz, and for SK506 at http://data.kaizhao.net/projects/skeleton/sk506.tar.gz. The chest X-ray data can be obtained from https://github.com/muhammedtalo/COVID-19. The DeepGlobe Road Extraction Challenge dataset is available at https://competitions.codalab.org/competitions/18467. The TuSimple Lane Detection Challenge is available at https://github.com/TuSimple/tusimple-benchmark/issues/3.

Code availability

The programming code used in this study is not publicly available due to it compromising intellectual property of Siemens Healthcare GmbH. The dataset-agnostic code for generation of 2D and 3D heatmaps of this study are available from the corresponding author upon reasonable request.

References

1. Shi, T., Kong, J.-Y., Wang, X.-D., Liu, Z. & Zheng, G. Improved Sobel algorithm for defect detection of rail surfaces with enhanced efficiency and accuracy. *J. Cent. South Univ.* **23**, 2867–2875. https://doi.org/10.1007/s11771-016-3350-3 (2016).
2. Tabernik, D., Šela, S., Skvarč, J. & Skočaj, D. Segmentation-based deep-learning approach for surface-defect detection. *J. Intell. Manuf.* **31**, 759–776. https://doi.org/10.1007/s10845-019-01476-x (2020).
3. Li, M., Zhang, Z., Xie, G. & Yu, J. A deep learning approach for face hallucination guided by facial boundary responses. *ACM Trans. Multimed. Comput. Commun. Appl.* **16**, 1–23. https://doi.org/10.1145/3377874 (2020).
4. Liu, Y. *et al.* A new dataset and boundary-attention semantic segmentation for face parsing. *AAAI Conf. Artif. Intell.* **34**, 11637–11644. https://doi.org/10.1609/aaai.v34i07.6832 (2020).
5. Wu, W., Zhang, Y., Li, C., Qian, C. & Loy, C. C. ReenactGAN: Learning to reenact faces via boundary transfer. *Comput. Vis.* **11205**, 622–638. https://doi.org/10.1007/978-3-030-01246-5_37 (2018).
6. Xie, J., Wan, J., Shen, L. & Lai, Z. Think about boundary: Fusing multi-level boundary information for landmark heatmap regression. *arXiv e-prints* arXiv:2008.10924 (2020).
7. Mnih, V. & Hinton, G. E. Learning to detect roads in high-resolution aerial images. In *Computer Vision (ECCV)* (eds Daniilidis, K. *et al.*) 210–223 (Springer, 2010).
8. Tu, Y., Wang, N., Tong, F. & Chen, H. Automatic measurement algorithm of scoliosis Cobb angle based on deep learning. *J. Phys.* **1187**, 042100. https://doi.org/10.1088/1742-6596/1187/4/042100 (2019).
9. Graovac, S. & Goma, A. Detection of road image borders based on texture classification. *Int. J. Adv. Robot. Syst.* **9**, 242. https://doi.org/10.5772/54359 (2012).
10. Philion, J. Fastdraw: Addressing the long tail of lane detection by adapting a sequential prediction network. In *Conf. Comput. Vision Pattern Recognit. (CVPR)* 11574–11583. https://doi.org/10.1109/CVPR.2019.01185 (2019).

11. Cheng, D. *et al.* Calving Front Machine (CALFIN): Glacial termini dataset and automated deep learning extraction method for greenland, 1972–2019. *The Cryosphere* **15**, 1663–1675. https://doi.org/10.5194/tc-15-1663-2021 (2021).
12. Zhang, J., Jia, L., Menenti, M. & Hu, G. Glacier facies mapping using a machine-learning algorithm: The Parlung Zangbo basin case study. *Remote Sens.* **11**, 452. https://doi.org/10.3390/rs11040452 (2019).
13. Yi, X., Adams, S., Babyn, P. & Elnajmi, A. Automatic catheter and tube detection in pediatric X-ray images using a scale-recurrent network and synthetic data. *J. Digit. Imaging* **33**, 181–190. https://doi.org/10.1007/s10278-019-00201-7 (2020).
14. Duda, R. O. & Hart, P. E. *Pattern Classification and Scene Analysis* (Willey, 1973).
15. Marr, D. & Hildreth, E. Theory of edge detection. *R. Soc. Lond. Ser. B* **207**, 187–217. https://doi.org/10.1098/rspb.1980.0020 (1980).
16. Prewitt, J. M. Object enhancement and extraction. *Pict. Process. Psychopict.* **10**, 15–19 (1970).
17. Canny, J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **PAMI–8**, 679–698. https://doi.org/10.1109/TPAMI.1986.4767851 (1986).
18. Kass, M., Witkin, A. & Terzopoulos, D. Snakes: Active contour models. *Int. J. Comput. Vis.* **1**, 321–331. https://doi.org/10.1007/BF00133570 (1988).
19. Cootes, T., Taylor, C., Cooper, D. & Graham, J. Active shape models: Their training and application. *Comput. Vis. Imaging Understand.* **61**, 38–59. https://doi.org/10.1006/cviu.1995.1004 (1995).
20. Edwards, G., Taylor, C. & Cootes, T. Interpreting face images using active appearance models. In *Third IEEE Int. Conf. Autom. Face Gesture Recognit.* 300–305 (IEEE Comput. Soc, 1998). , https://doi.org/10.1109/AFGR.1998.670965.
21. Tompson, J., Jain, A., LeCun, Y. & Bregler, C. Joint training of a convolutional network and a graphical model for human pose estimation. In *27th Int. Conf. Neural Inf. Process. Syst.* 1799-1807 (MIT Press, 2014).
22. Yu, B. & Tao, D. Heatmap Regression via Randomized Rounding. *arXiv e-prints* arXiv:2009.00225 *(2020)*.
23. Chen, Y., Tian, Y. & He, M. Monocular human pose estimation: A survey of deep learning-based methods. *Comput. Vis. Imaging Understand.* **192**, 102897. https://doi.org/10.1016/j.cviu.2019.102897 (2020).
24. Neven, D., Brabandere, B. D., Georgoulis, S., Proesmans, M. & Gool, L. V. Towards end-to-end lane detection: An instance segmentation approach. *Intell. Veh. Symp.* **286–291**, 2018. https://doi.org/10.1109/IVS.2018.8500547 (2018).
25. Adhikari, S. P. & Kim, H. Distance weighted loss for forest trail detection using semantic line. *Adv. Concepts Intell. Vis. Syst.* **12002**, 302–311. https://doi.org/10.1007/978-3-030-40605-9_26 (2020).
26. Holzmann, M. *et al.* Glacier calving front segmentation using attention u-net. *arXiv e-prints* arXiv:2101.03247 *(2021)*.
27. Deng, R., Shen, C., Liu, S., Wang, H. & Liu, X. Learning to predict crisp boundaries. *Comput. Vis.* **11210**, 570–586. https://doi.org/10.1007/978-3-030-01231-1_35 (2018).
28. Kervadec, H. *et al.* Boundary loss for highly unbalanced segmentation. *Med. Image Anal.* **67**, 101851. https://doi.org/10.1016/j.media.2020.101851 (2021).
29. Blum, H. A transformation for extracting new descriptors of shape. In *Models for the Perception of Speech and Visual Form* (ed. Wathen-Dunn, W.) 362–380 (MIT Press, 1967).
30. Cornea, N. D., Silver, D. & Min, P. Curve-skeleton applications. In *VIS 05. IEEE Vis., 2005.* 95–102 (2005).
31. Tsogkas, S. & Kokkinos, I. Learning-based symmetry detection in natural images. In *Computer Vision (ECCV)* (eds Fitzgibbon, A. *et al.*) 41–54 (Springer, 2012).
32. Sironi, A., Lepetit, V. & Fua, P. Multiscale centerline detection by learning a scale-space distance transform. In *Conf. Comput. Vision Pattern Recognit. (CVPR)* 2697–2704. https://doi.org/10.1109/CVPR.2014.351 (2014).
33. Shen, W. *et al.* Deepskeleton: Learning multi-task scale-associated deep side outputs for object skeleton extraction in natural images. *IEEE Trans. Imaging Process.* **26**, 5298–5311. https://doi.org/10.1109/TIP.2017.2735182 (2017).
34. Zhao, K., Shen, W., Gao, S., Li, D. & Cheng, M. Hi-Fi: Hierarchical feature integration for skeleton detection. In Lang, J. (ed.) *Int. Joint Conf. Artif. Intell.* 1191–1197. https://doi.org/10.24963/ijcai.2018/166 (2018).
35. Xu, Y. *et al.* Deepflux for skeleton detection in the wild. *Int. J. Comput. Vis.* **129**, 1323–1339. https://doi.org/10.1007/s11263-021-01430-6 (2021).
36. Payer, C., Štern, D., Bischof, H. & Urschler, M. Regressing heatmaps for multiple landmark localization using cnns. In *Med. Imag. Comput. Comput. Assist. Intervent. MICCAI* (eds Ourselin, S. *eta l.*) 230–238 (Springer, 2016).
37. Wei, S., Ramakrishna, V., Kanade, T. & Sheikh, Y. Convolutional pose machines. In *2016 Conf. Comput. Vision Pattern Recognit. (CVPR)* 4724–4732. https://doi.org/10.1109/CVPR.2016.511 (2016).
38. Chandran, P., Bradley, D., Gross, M. & Beeler, T. Attention-driven cropping for very high resolution facial landmark detection. In *Conf. Comput. Vision Pattern Recognit. (CVPR)* 5860–5869, https://doi.org/10.1109/CVPR42600.2020.00590 (2020).
39. Qu, Z., Jin, H., Zhou, Y., Yang, Z. & Zhang, W. Focus on local: Detecting lane marker from bottom up via key point. In *Conf. Comput. Vision Pattern Recognit. (CVPR)* 14122–14130 (2021).
40. Navarro, F. *et al.* Shape-aware complementary-task learning for multi-organ segmentation. In *Machine Learning for Medical Imaging* (eds Suk, H.-I. *et al.*) 620–627 (Springer, Cham, 2019).
41. Wu, W. *et al.* Look at boundary: A boundary-aware face alignment algorithm. In *Conf. Comput. Vision Pattern Recognit. (CVPR)* 2129–2138. https://doi.org/10.1109/CVPR.2018.00227 (2018).
42. Kan, J. *et al.* Sign language translation with hierarchical spatio-temporal graph neural network. In *IEEE Winter Conf. Appl. Comput. Vis. (WACV)* 3367–3376 (2022).
43. Li, W. *et al.* Structured landmark detection via topology-adapting deep graph learning. In *Computer Vision (ECCV)*, vol. 12354 of *LCNS* (eds Vedaldi, A. *et al.*) 266–283 (Springer, 2020). https://doi.org/10.1007/978-3-030-58545-7_16.
44. Gaggion, N., Mansilla, L., Mosquera, C., Milone, D. H. & Ferrante, E. Improving anatomical plausibility in medical image segmentation via hybrid graph neural networks: Applications to chest x-ray analysis. *arXiv e-prints* arXiv:2203.10977 *(2022)*.
45. Yu, W. *et al.* Layout-graph reasoning for fashion landmark detection. In *Conf. Comput. Vision Pattern Recognit. (CVPR)* (2019).
46. Arbelaez, P., Maire, M., Fowlkes, C. & Malik, J. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**, 898–916 (2011).
47. Silberman, N., Hoiem, D., Kohli, P. & Fergus, R. Indoor segmentation and support inference from rgbd images. In *Computer Vision (ECCV))* (eds Fitzgibbon, A. *et al.*) 746–760 (Springer, 2012).
48. Shen, W., Bai, X., Hu, Z. & Zhang, Z. Multiple instance subspace learning via partial random projection tree for local reflection symmetry in natural images. *Pattern Recogn.* **52**, 306–316 (2016).
49. Shen, W. *et al.* Object skeleton extraction in natural images by fusing scale-associated deep side outputs. In *Conf. Comput. Vision Pattern Recognit. (CVPR)* 222–230 (2016).
50. No author given. TuSimple Lane Detection Challenge. https://github.com/TuSimple/tusimple-benchmark (2021).
51. Demir, I. *et al.* Deepglobe 2018: A challenge to parse the earth through satellite images. In *Conf. Comput. Vision Pattern Recognit. (CVPR)* (2018).
52. Mitchell, T. M. *Machine Learning* (McGraw-Hill, 1997).
53. Dierckx, P. Algorithms for smoothing data with periodic and parametric splines. *Comput. Graph. Imag. Process.* **20**, 171–184. https://doi.org/10.1016/0146-664X(82)90043-0 (1982).
54. Reinsch, C. H. Smoothing by spline functions. *Numer. Math.* **10**, 177–183. https://doi.org/10.1007/BF02162161 (1967).
55. Wang, H., Kearney, J. & Atkinson, K. Arc-length parameterized spline curves for real-time simulation. In *Int. Conf. Curves and Surfaces* 387–396 (2002).

56. Floater, M. S., Rasmussen, A. F. & Reif, U. Extrapolation methods for approximating arc length and surface area. *Numer. Algorithms* **44**, 235–248. https://doi.org/10.1007/s11075-007-9095-1 (2007).
57. Balta, C., Öztürk, S., Kuncan, M. & Kandilli, I. Dynamic centripetal parameterization method for b-spline curve interpolation. *IEEE Access* **8**, 589–598. https://doi.org/10.1109/ACCESS.2019.2961412 (2020).
58. Floater, M. S. & Rasmussen, A. F. Point-based methods for estimating the length of a parametric curve. *J. Comput. Appl. Math.* **196**, 512–522. https://doi.org/10.1016/j.cam.2005.10.001 (2006).
59. Lee, E. Choosing nodes in parametric curve interpolation. *Comput.-Aided Des.* **21**, 363–370. https://doi.org/10.1016/0010-4485(89)90003-1 (1989).
60. Shepard, D. A two-dimensional interpolation function for irregularly-spaced data. In *23rd ACM National Conf.* ACM '68, 517–524 (Association for Computing Machinery, 1968). https://doi.org/10.1145/800186.810616.
61. Maurer, C., Qi, R. & Raghavan, V. A linear time algorithm for computing exact Euclidean distance transforms of binary images in arbitrary dimensions. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**, 265–270. https://doi.org/10.1109/TPAMI.2003.1177156 (2003).
62. Amis, A. A. & Jakob, R. P. Anterior cruciate ligament graft positioning, tensioning and twisting. *Knee Surg. Sports Traumatol. Arthrosc.* **6**(Suppl 1), S2-12. https://doi.org/10.1007/s001670050215 (1998).
63. Bernard, M., Hertel, P., Hornung, H. & Cierpinski, T. Femoral insertion of the ACL radiographic quadrant method. *Am. J. Knee Surg.* **10**, 14–21 (1997) (**discussion 21–2**).
64. Laube, P., Franz, M. O. & Umlauf, G. Deep learning parametrization for B-spline curve approximation. In *2018 International Conference on 3D Vision (3DV)* 691–699. https://doi.org/10.1109/3DV.2018.00084 (2018).
65. Sezgin, M. & Sankur, B. Survey over image thresholding techniques and quantitative performance evaluation. *J. Electron. Imag.* **13**, 146–165. https://doi.org/10.1117/1.1631315 (2004).
66. Zhang, T. Y. & Suen, C. Y. A fast parallel algorithm for thinning digital patterns. *Commun. ACM* **27**, 236–239. https://doi.org/10.1145/357994.358023 (1984).
67. Akkus, Z., Galimzianova, A., Hoogi, A., Rubin, D. L. & Erickson, B. J. Deep learning for brain MRI segmentation: State of the art and future directions. *J. Digit. Imaging* **30**, 449–459. https://doi.org/10.1007/s10278-017-9983-4 (2017).
68. Dollár, P. & Zitnick, C. L. Fast edge detection using structured forests. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**, 1558–1570 (2014).
69. Xie, S. & Tu, Z. Holistically-nested edge detection. In *IEEE Conf. Comput. Vision (ICCV)* 1395–1403 (2015).
70. Newell, A., Yang, K. & Deng, J. Stacked hourglass networks for human pose estimation. In *Computer Vision (ECCV)* (eds Leibe, B. *et al.*) 483–499 (Springer, 2016).
71. Kordon, F. *et al.* Multi-task localization and segmentation for X-ray guided planning in knee surgery. *Med. Imag. Comput. Comput. Assist. Intervent.* **11769**, 622–630. https://doi.org/10.1007/978-3-030-32226-7_69 (2019).
72. He, K., Zhang, X., Ren, S. & Sun, J. Identity mappings in deep residual networks. In *Computer Vision (ECCV)* (eds Leibe, B. *et al.*) 630–645 (Springer, 2016).
73. He, K., Zhang, X., Ren, S. & Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *IEEE Conf. Comput. Vision (ICCV)* 1026–1034. https://doi.org/10.1109/ICCV.2015.123 (2015).
74. Paszke, A. *et al.* PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing System* (eds Wallach, H. *et al.*) 8026–8037 (Curran Associates, Inc., 2019).
75. Ozturk, T. *et al.* Automated detection of COVID-19 cases using deep neural networks with X-ray images. *Comput. Biol. Med.* **121**, 103792. https://doi.org/10.1016/j.compbiomed.2020.103792 (2020).
76. Cohen, J. P. *et al.* COVID-19 Image Data Collection: Prospective predictions are the future. arXiv:2006.11988 (2020).
77. Wang, X. *et al.* ChestX-Ray8: Hospital-scale chest X-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Conf. Comput. Vision Pattern Recognit. (CVPR)* 3462–3471 (IEEE, 2017). https://doi.org/10.1109/CVPR.2017.369.
78. Wada, K. labelme: Image polygonal annotation with Python (2016).
79. Meyer, E., Raupach, R., Lell, M., Schmidt, B. & Kachelrieß, M. Frequency split metal artifact reduction (FSMAR) in computed tomography. *Med. Phys.* **39**, 1904–1916. https://doi.org/10.1118/1.3691902 (2012).
80. Lee, K., Zung, J., Li, P., Jain, V. & Seung, H. S. Superhuman accuracy on the SNEMI3D Connectomics Challenge. *arXiv e-print-sarXiv:1706.00120 (2017).*
81. Gupta, S., Arbelaez, P. & Malik, J. Perceptual organization and recognition of indoor scenes from RGB-D images. In *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 564–571 (2013).

Author contributions
F.K., M.S., and H.K. designed the theoretical formalism and did the study design. F.K., H.K., and C.V. designed the computational framework. F.K. implemented the framework. C.V. and T.P. extended the method to 3D. F.K. implemented and executed the property analysis and experiments on 2D datasets. F.K. and T.P. executed the 3D experiments. F.K. analyzed the results and performed the experiment evaluation. H.K. and A.M. received the funding and supervised the study. The manuscript was written by F.K., M.S., and H.K. All authors corrected and approved the paper.

Funding

Competing interests

Additional information
Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-023-41221-2.

Correspondence and requests for materials should be addressed to F.K.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.