Journal of Mathematical Imaging and Vision
https://doi.org/10.1007/s10851-020-00964-7

Stochastic Distance Transform: Theory, Algorithms and Applications

Johan Öfverstedt[1] · **Joakim Lindblad**[1] · **Nataša Sladoje**[1]

Received: 9 July 2019 / Accepted: 7 May 2020
© The Author(s) 2020

Abstract
Distance transforms (DTs) are standard tools in image analysis, with applications in image registration and segmentation. The DT is based on extremal (minimal) distance values and is therefore highly sensitive to noise. We present a *stochastic distance transform* (SDT) based on *discrete random sets*, in which a model of element-wise probability is utilized and the SDT is computed as the first moment of the distance distribution to the random set. We present two methods for computing the SDT and analyze them w.r.t. accuracy and complexity. Further, we propose a method, utilizing kernel density estimation, for estimating probability functions and associated random sets to use with the SDT. We evaluate the accuracy of the SDT and the proposed framework on images of thin line structures and disks corrupted by salt and pepper noise and observe excellent performance. We also insert the SDT into a segmentation framework and apply it to overlapping objects, where it provides substantially improved performance over previous methods. Finally, we evaluate the SDT and observe very good performance, on simulated images from localization microscopy, a state-of-the-art super-resolution microscopy technique which yields highly spatially localized but noisy point-clouds.

Keywords Distance transform · Discrete random set · Robust distance · Image segmentation · Deterministic algorithm · Monte-Carlo method

1 Introduction

Distance transforms (DTs) [6,7,46,47] are standard tools in image analysis and processing and are used for a wide range of tasks such as image registration [8,20,44], template matching [37], classification [37], segmentation [5] and skeletonization [49]. In many scenarios when DTs are used, the input image may be noisy or the processing step preceding the DT may yield an unreliable output, manifested as noise. DTs are based on extremal (minimal) distance values and are therefore very sensitive to noise; robust estimation of the DT is a challenging issue which is only partly addressed in the literature.

Discrete random sets DRSs [25,26,39] are random variables taking values as subsets of some reference space. Modeling noisy (corrupted) binary images as DRSs lets us apply a rich set of probabilistic tools to analyze and mitigate the impact of the noise. In the theory of DRSs, the *coverage probability function* CPF of a DRS gives the probability of each element being included in the set, which enables probabilistic reasoning of the individual elements of the set.

One method for computing a robust Hausdorff distance between noisy sets of points, utilizing the distance transform, is based on random removal of points to compute weighted distance histograms based on user-provided point-wise reliability coefficients [42]. Continuing this research track on how the accuracy of DTs can be improved, starting from the theory of DRSs rather than distance histograms, we propose a *Stochastic Distance Transform* (SDT). The SDT is based on the stochastic expectation of minimal point-to-set distances in noisy images. An initial study of a special case of this concept was introduced in [45], establishing the ability of this approach to reduce the impact of additive noise on the DT. In this extended work, we develop and analyze a more general and flexible framework. We also present and investigate a method, based on kernel density estimation (KDE), to estimate the coverage probability function. We evaluate the performance of the proposed framework on corrupted images (with both additive and subtractive noise), both in terms of

This work is supported by VINNOVA, MedTech4Health Grants 2016-02329 and 2017-02447. Funding was provided by Ministarstvo Prosvete, Nauke i Tehnološkog Razvoja (Grant Nos. ON174008 and III44006).

✉ Johan Öfverstedt
johan.ofverstedt@it.uu.se

[1] Department of Information Technology, Uppsala University, Lägerhyddsvägen 2, Box 337, 751 05 Uppsala, Sweden

Published online: 19 June 2020

accuracy of the computed DT values and its performance when utilized for image segmentation.

A number of theoretical frameworks for computing robust distances between sets and distributions have been proposed. One such framework relies on distances based on optimal transport (OT), which models the amount of displacement of mass required to transform one distribution into another, subject to an underlying spatial distribution [2,21,55]. Optimal transport problems (and related distances) suffer from a heavy computational cost that is prohibitive in large-scale applications. Recent approximate solutions, building on the use of Sinkhorn iterations [21], reduce the computation load substantially, however, at the cost of numerical stability challenges. Distances based on optimal transport may be inherently unstable and noise sensitive; regularization techniques, e.g., based on entropy, can mitigate this issue [2], if a certain level of smoothing is acceptable in the computed approximation. OT typically considers the transformation of one distribution into another and is as such not directly applicable as a point-to-set distance (or DT).

One research track towards a robust point-to-set distance, and the reconstruction of geometric objects subject to noise is [12]. Instead of assuming an underlying compact set that is sampled by the points, they assume an underlying probability measure from which a point sample is drawn. The distance function to the measure depends on a mass parameter which acts as a smoothing term. This work is further developed into the k-distance [18,27] which provides a practical algorithm for a noise-insensitive distance measure by utilizing an (unweighted) aggregation of the distances to the k nearest points. The here-proposed deterministic method for computing the SDT resembles a refined weighted version of the k-distance. In this work, we observe that, by allowing distinct weights of the aggregated distances, the proposed SDT becomes a more general tool, which can treat both subtractive and additive noise successfully.

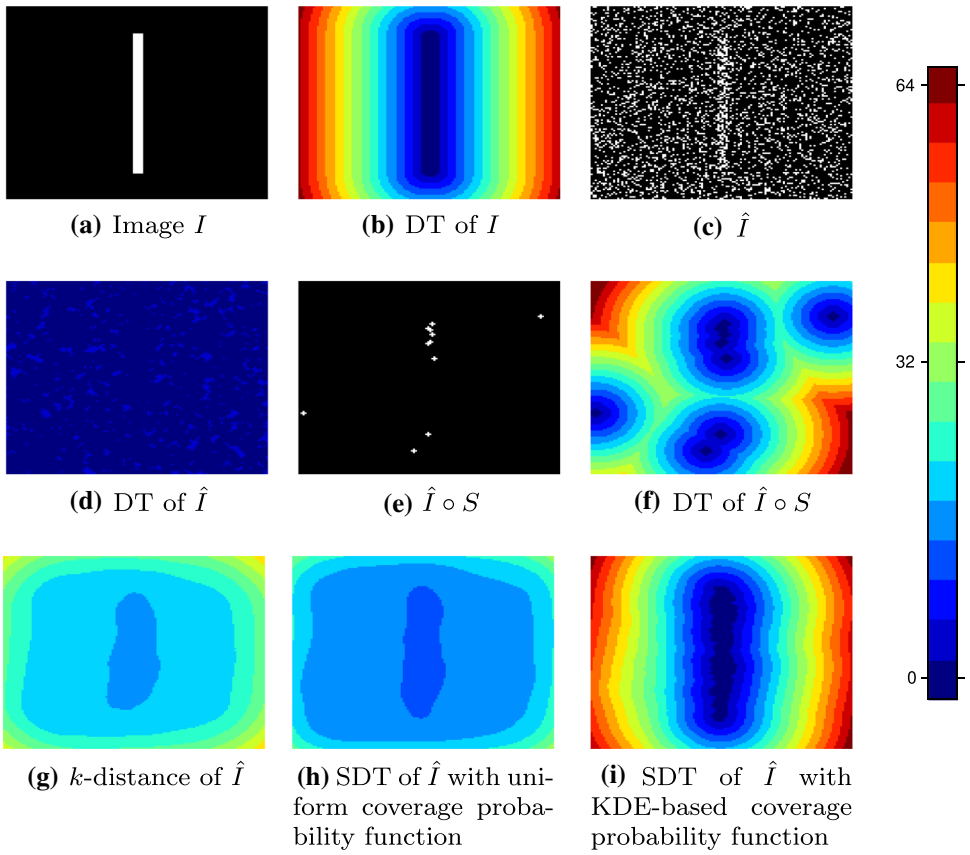
Persistent homology [13,14,22,64] is another rich framework for extraction of multi-scale noise-robust topological structures. By observing the stability of the topology as a function of increasing size of, e.g., disks growing from each point, topologically salient (noise-insensitive) representations of the raw data can be obtained. The resulting persistence diagrams are useful for comparing images with respect to topology but do not directly provide geometric information. In a very recent work, [10] is shown a non-trivial relation between persistent homology and curvature of an object. The authors propose to learn this relation using standard tools from statistical and machine learning.

Figure 1 provides an illustrative example of the challenge that noise imposes on the accuracy of the DT and different ways to address it. It can be difficult to formulate a rule for removing noise and filling holes such that all noise is successfully removed and all holes are filled. Considering that the DT is based on distances to the nearest point of a set, any remaining noise can corrupt large regions of the DT, while incorrectly removed object points can lead to overestimations of distances and loss of structural information. Figure 1e shows the result of applying a morphological opening (a common denoising method) on the noisy image, which has (i) failed to remove all noise and (ii) removed substantial parts of the object of interest. Figure 1f shows the DT of the image after opening, which displays severe corruption compared to the desired result in Fig. 1b. Figure 1g, h shows the DT resulting from the k-distance as well as the proposed SDT method with a uniform certainty map [45]. The two results are similar; the impact of noise is reduced compared to the results of using a classic DT (Fig. 1d, f), but the distances are substantially underestimated. Figure 1i shows the DT resulting from the here-proposed method with a non-uniform coverage probability function (estimated by a Kernel Density Estimator), which visibly corresponds well to the noise-free desired output in Fig. 1b.

2 Contributions of the Paper

This work extends [45], where we introduced the SDT, a mathematical tool based on the theory of DRSs, for computing robust distances in the presence of noise. In the here-presented study, we include general definitions of the transform as well as two methods for its efficient computation: one based on Monte-Carlo simulation and one based on ordered sequences of nearest points. We perform analysis of the approximation accuracy and computational complexity for both methods. We expand the discussion of advantages and limitations of the special case of the SDT presented in [45], where coverage probability functions are restricted to be uniform and incorporate this case into the more general version of the framework. Further, we propose a method, based on KDE, to estimate coverage probability functions from noisy input images.

We perform empirical analysis of the accuracy of the DT in the presence of both additive and subtractive (salt and pepper) noise, comparing performances in combination with a number of common kernels for density estimation. In addition, we evaluate the SDT on localization microscopy images which are generated by a software simulator [43,53], providing known ground-truth and therefore direct evaluation while still considering realistic data. We observe that the proposed method has excellent performance compared to several other candidate methods. We also insert the SDT into an image segmentation framework based on watersheds and evaluate the robustness of segmentation of partially overlapping objects in the presence of border noise and discretization artifacts.



(a) Image I **(b)** DT of I **(c)** $\hat{I}$

(d) DT of $\hat{I}$ **(e)** $\hat{I} \circ S$ **(f)** DT of $\hat{I} \circ S$

(g) k-distance of $\hat{I}$ **(h)** SDT of $\hat{I}$ with uniform coverage probability function **(i)** SDT of $\hat{I}$ with KDE-based coverage probability function

Fig. 1 Impact of noise on the DT. **a** Example image with a single line segment; **b** DT of **a**; **c** the image **a** corrupted by salt and pepper noise; **d** the DT of the noisy image **c**; **e** the noisy image after a morphological opening with a disk of radius 1; **f** the DT of the filtered image in **e**; **g** the k-distance of **c**; **h** the SDT of **c** with a uniform coverage probability function [45]; and **i** the result of the here proposed SDT of **c**, based on kernel density estimation, exhibiting only small amounts of corruption caused by the noise (to be compared with **b**)

3 Background and Previous Work

In this work, we rely on existing concepts and results related to random sets, distance transforms, kernel density estimation, fuzzy set theory, and the k-distance.

3.1 Discrete Random Sets

Discrete random sets (DRSs) [25,26,39] are random variables taking values as subsets of some discrete reference space. A DRS $\mathcal{Y}$ on a domain (reference space) I is characterized by its probability-mass function (PMF),

$$f_{\mathcal{Y}}(X) = \mathbb{P}(\mathcal{Y} = X),$$

where $X \subseteq I$ is a standard (non-random) set. DRSs have been shown to be useful in image processing [1], especially to model (random) binary images which can directly be related through binary random fields [25]. Theory of DRSs provides foundation and offers suitable tools for the modeling and analysis of shapes in images, allowing exploration of both their structural and statistical characteristics. In particular, representing (binary) image objects as (finite) random subsets on an image domain I (bounded on $\mathbb{Z}^n$) facilitates their structural analysis in presence of noise.

The coverage probability function (CPF) [1,35] of a DRS $\mathcal{Y}$ expresses the probability that $\mathcal{Y}$ contains x and is defined as

$$p_{\mathcal{Y}}(x) = \mathbb{P}(x \in \mathcal{Y}). \tag{1}$$

If the inclusions of the elements of the domain in a DRS $\mathcal{Y}$ are mutually independent, then the CPF uniquely defines the DRS, with the following PMF:

$$f_{\mathcal{Y}}(X) = \prod_{x \in X} p_{\mathcal{Y}}(x) \cdot \prod_{x \in I \setminus X} (1 - p_{\mathcal{Y}}(x)).$$

The *avoidance functional* [41] of a DRS and a subset $K \subseteq I$ is

$$Q_{\mathcal{Y}}(K) = \mathbb{P}(\mathcal{Y} \cap K = \emptyset). \tag{2}$$

The void probability [25] is the probability that a DRS takes on the empty set, $f_{\mathcal{Y}}(\emptyset) = \mathbb{P}(\mathcal{Y} = \emptyset)$.

3.2 Distance Transforms

The initial works on distance transforms focused on raster-scan methods on binary images for computing Manhattan and Chessboard distances [46,47], followed by more refined

weighted (Chamfer mask) distance transforms [7,15] which approximate the Euclidean distance transform (EDT) with smaller error than previous methods. Fast, separable algorithms have since then been developed which eliminate approximation errors and compute the exact Euclidean distance transform in linear time [16,24,40] and which are parallelizable and fast in practice.

There are two major themes in the DT-related literature: (i) methods that introduce more sophisticated models for the shape and location of the objects to which the distance is being computed and (ii) methods that consider different media in which the distances are propagated through.

Works belonging to category (i) include the anti-aliased distance transform [28,30], which improves the sub-pixel error of distances to continuous structures (discretized in a grid) and the exact EDT for grid sampled signals [38]. Another study investigates models for imprecision in the precise location of digital objects [54]. A different but related aspect to this, concerning distance transforms defined on irregular grids, was explored in [56–58]. The SDT proposed in this work also belongs to category (i).

Works belonging to category (ii) include the gray-scale and fuzzy distance transform methods [29,34,50] which use available information-rich representations to obtain more accurate distance transforms.

Definitions of distances between points and sets commonly build on a distance measure between points. The Euclidean point-to-point distance d_E is a natural and common choice and is used as the default in this study, unless explicitly specified. Note that the framework discussed in this study does not rely on metric properties of the distances, though explicit methods for computation may.

Given a point-to-point distance d, the standard point-to-set distance between a point x and a set Y is defined as

$$d(x, Y) = \inf_{y \in Y} d(x, y). \tag{3}$$

The (unsigned, external) distance transform (DT) with respect to a set (image object) Y evaluated on the image domain I is

$$\mathtt{DT}[Y](x) = \min_{y \in Y} d(x, y), \qquad \text{for} \quad x \in I. \tag{4}$$

3.3 Distances to Discrete Random Sets

Point-to-set distances can also be extended to DRSs [1] and hence generate probability distributions of distance values. The distance $d(x, \mathcal{Y})$ is a random variable with values $[0, \infty]$, taking on the value ∞ when $\mathcal{Y} = \emptyset$.

If $d(x, \mathcal{Y})$ is integrable for all $x \in I$, (i.e., the void probability is 0 for distances which take on the value ∞ for empty sets), then a mean distance function for DRSs [1,35] can be defined as

$$\bar{d}(x) = \mathbb{E}d(x, \mathcal{Y}). \tag{5}$$

The requirement of a zero void probability is too restrictive for the context of this work, since it is reasonable to consider scenarios where the presence of no single element is completely certain. By saturating the distance to a maximum value $d_{\mathtt{MAX}}$, the mean distances are well-defined and finite even for empty sets.

Given a set of realizations of a DRS, $Y_1, Y_2, Y_3 \ldots Y_n$, an empirical sample mean (or some other statistics such as median or sample variance) of the distances $d(x, Y_1), \ldots, d(x, Y_n)$ can be computed, [1,35]. This is a useful tool when our input consists of multiple images, representing realizations of the same underlying DRS [1], or when we are able to sample images from a DRS, which forms the basis of the proposed Monte-Carlo method. In this work, we propose methods for efficiently computing expected distances to DRSs under specific assumptions, such that we can compute noise-robust DTs given a single binary image.

3.4 The *k*-Distance

Let $n_Y^{(i)}(x)$ denote the i-th point nearest to x in Y (if multiple points are equidistant to x, the tie is broken arbitrarily), and let $\mathcal{N}_Y^{(k)}(x)$ denote the set of the k nearest points to x in Y, $\mathcal{N}_Y^{(k)}(x) = \{n_Y^{(1)}(x), \ldots, n_Y^{(k)}(x)\}$.

The k-distance [18,27] is the (unweighted) root-mean-square of the distances from the k nearest points $n_Y^{(i)}(x)$ to x:

$$d_k[Y](x) = \sqrt{\frac{1}{k} \sum_{i=1}^{k} d^2(x, n_Y^{(i)}(x))}. \tag{6}$$

The k-distance exhibits a degree of robustness to additive noise where the appearance of a few noise points have a limited impact on the average. However, a remedy for subtractive (pepper) noise is not offered, since points incorrectly not included (due to noise) in the set Y remain excluded from the mean of Eq. (6). The k-distance exhibits a bias inside dense objects where the distance is overestimated; it takes on a mean of the distances to the k nearest grid-points, which is greater than zero.

3.5 Fuzzy Set Theory and Distances

We recall a few basic concepts related to fuzzy sets [62], a theoretical framework for representing partial membership/belongingness of elements to sets/objects. Fuzzy set theory is one of the major theoretical frameworks that enables

encoding of precision and is the foundation of many noise-robust methods.

A *fuzzy set* $\mathcal{S}$ on a reference set $X_\mathcal{S}$ is a set of ordered pairs, $\mathcal{S} = \{(x, \mu_\mathcal{S}(x)) : x \in X_\mathcal{S}\}$, where $\mu_\mathcal{S} \colon X_\mathcal{S} \to [0, 1]$ is the *membership function* of $\mathcal{S}$.

A crisp set $C \subseteq X_C$ (a binary image) is a special case of a fuzzy set, with its characteristic function as membership function

$$\mu_C(x) = \begin{cases} 1, & \text{for } x \in C \\ 0, & \text{for } x \notin C\,. \end{cases} \tag{7}$$

The *height* of a fuzzy set $S \subseteq X_\mathcal{S}$ is $h(\mathcal{S}) = \max_{x \in X_\mathcal{S}} \mu_\mathcal{S}(x)$. An α-*cut* of a fuzzy set $\mathcal{S}$ is a crisp set defined as ${}^\alpha\mathcal{S} = \{x \in X_\mathcal{S} : \mu_\mathcal{S}(x) \geq \alpha\}$, i.e., a thresholded image.

A fuzzy point-to-set distance measure [37] which has shown excellent performance on a number of tasks such as template matching and classification is given by the integral over α-cuts,

$$d^\alpha(x, \mathcal{S}) = \int_0^{h(x)} d(x, {}^\alpha\mathcal{S})\, \mathrm{d}\alpha, \tag{8}$$

where $h(x)$ is the height of point x, and $h(x) \in [0, 1]$. In this work, all heights are unit-valued, since we compute distances from the (crisp) domain.

Fuzzy sets and element-wise independent DRSs are superficially similar in that fuzzy sets associate each element with a partial membership to a reference set, while an element-wise independent DRS associates each element with a CP. However, the meanings of the two concepts differ substantially, since a fuzzy set is not a random variable [62], but rather a tool for encoding deterministic partial inclusion in a set.

3.6 Kernel Density and Mass Estimation

Kernel density estimation (KDE) [23,31,51] is a nonparametric method where a (weighted) kernel, or window function, is used to estimate the probability density function of a continuous random variable or probability mass function of a discrete random variable.

Methods based on KDE have been successfully applied in denoising of noisy point-clouds [9,63], clustering (of noisy point-cloud data) [17] and different types of outlier detection [33,36], which are all based on separating salient elements from non-salient elements based on density.

There are many KDE kernel functions in common use [32]; a subset of those defined for univariate kernels are listed in Table 1, all of which have a finite support on $|z| \leq 1$. There are multiple ways of transforming such univariate kernels into multivariate kernels. Given a univariate kernel K and a suitable norm, a *radially symmetric* multivariate kernel [32,51] with radius r is given by

Table 1 Common univariate kernels

Uniform	$K_\mathrm{U}(z) = \frac{1}{2}$
Triangle	$K_\mathrm{T}(z) = 1 - \lvert z\rvert$
Biweight	$K_\mathrm{BW}(z) = \frac{15}{16}(1 - \lvert z\rvert^2)^2$
Triweight	$K_\mathrm{TW}(z) = \frac{35}{32}(1 - \lvert z\rvert^2)^3$
Epanechnikov [23,31]	$K_\mathrm{E}(z) = \frac{3}{4}(1 - \lvert z\rvert^2)$
Cosine	$K_\mathrm{cos}(z) = \frac{\pi}{4}\cos(\frac{\pi z}{2})$

The listed kernels have a support $|z| \leq 1$ and take the value 0 outside of this interval

$$K_\circ(\mathbf{z}) = \beta K\left(\frac{\|\mathbf{z}\|}{r}\right), \tag{9}$$

where β is a normalization constant. For this study, when considering images on rectangular grid domains, the normalization constant is suitably chosen so that the kernel is normalized to a unit sum. For the discretized versions of the continuous kernels, the procedure becomes kernel mass estimation.

4 The Stochastic Distance Transform

4.1 Notation and Definitions

Let $\hat{Y} = f(Y)$ be an observed binary image on a domain I, resulting from a true image Y which is corrupted by a random field (stochastic process) f. In other words, the observed binary image $\hat{Y} = f(Y)$ can be seen as one sample of a discrete random set $\mathcal{Y}$ on I.

The DRS $\mathcal{Y}$ can be characterized by its CPF $p_\mathcal{Y}(y) = \mathbb{P}(y \in Y)$, $\forall y \in I$, which is either known or estimated. For all DRSs in this work, we assume mutual independence of inclusion of elements. This simplifies definitions and algorithms significantly and is also reasonable in many imaging scenarios.

We now introduce the SDT, a mathematical tool for computing the distance map on an image domain w.r.t. an object represented by a DRS $\mathcal{Y}$.

Definition 1 Given an image domain I, a DRS $\mathcal{Y}$ on I, a maximal distance $d_\mathtt{MAX} \in \mathbb{R}_+$, and a point-to-set distance d, the (unsigned, external) *Stochastic Distance Transform (SDT)* is

$$\mathtt{SDT}[\mathcal{Y}](x) = \mathbb{E}(\min[d(x, \mathcal{Y}), d_\mathtt{MAX}]), \quad \text{for } x \in I. \tag{10}$$

$d_\mathtt{MAX}$ is typically taken to be the diameter of the domain I. If $\mathcal{Y}$ has a nonzero void probability, the saturation to a max-

imal distance yields a finite and well-defined expectation in Definition 1.

We also define the variance of the point-to-set distance distribution underlying the SDT.

Definition 2 Given an image domain I, a DRS $\mathcal{Y}$ on I, a maximal distance $d_{\texttt{MAX}} \in \mathbb{R}_+$, and a point-to-set distance d, the (unsigned, external) *Stochastic Distance Variance Transform (SDVT)* is

$$\texttt{SDVT}\left[\mathcal{Y}\right](x) = \texttt{Var}(\min\left[d(x,\mathcal{Y}), d_{\texttt{MAX}}\right]), \quad \text{for } x \in I. \tag{11}$$

For scenarios where each element in the set of interest is associated with a coverage probability, from, e.g., a prior distribution of an imaging process, the SDT can be used directly without further pre-processing. If no such probabilities are readily available, they can be estimated using some established method, e.g., KDE.

For the further analysis of the SDT, we borrow some terminology from fuzzy set theory and adapt it to DRSs. The support of $\mathcal{Y}$ is

$$\text{supp}(\mathcal{Y}) = \left\{x : x \in I \texttt{ and } p_{\mathcal{Y}}(x) > 0\right\}. \tag{12}$$

The core of $\mathcal{Y}$ is

$$\text{core}(\mathcal{Y}) = \left\{x : x \in I \texttt{ and } p_{\mathcal{Y}}(x) = 1\right\}. \tag{13}$$

We denote the size of $\mathcal{Y}$ as the cardinality of the support:

$$|\mathcal{Y}| = |\text{supp}(\mathcal{Y})|. \tag{14}$$

The minimum nonzero probability of $p_{\mathcal{Y}}$ is denoted $\min_+(\mathcal{Y})$, which is equal to 0 only when $\text{supp}(\mathcal{Y}) = \emptyset$.

4.2 Monte Carlo Method

An estimate of $\texttt{SDT}\left[\mathcal{Y}\right](x)$ can be obtained by a Monte Carlo method, denoted MC-SDT, by drawing N random samples (sets) from $\mathcal{Y}$, $(Y_1, \ldots, Y_N)$, computing the corresponding DTs using a suitable algorithm and then computing their empirical mean,

$$\texttt{MC-SDT}_N\left[\mathcal{Y}\right](x) = \frac{1}{N}\sum_{i=1}^{N} \min\left[d(x, Y_i), d_{\texttt{MAX}}\right]. \tag{15}$$

Due to the mutual independence of inclusions of elements, each sample Y_i can be generated by one independent Bernoulli trial per element in $\text{supp}(\mathcal{Y})$.

4.2.1 Error Analysis

The Monte Carlo method is a sample mean estimator (with finite mean and variance); by the Central Limit Theorem, the margin of error is given by

$$ME = \frac{2\sqrt{\texttt{SDVT}\left[\mathcal{Y}\right](x)}}{\sqrt{N}}, \tag{16}$$

for sufficiently large N, and asymptotically the margin of error decreases as $\mathcal{O}(\frac{1}{\sqrt{N}})$. The error is dependent on $\mathcal{Y}$ (which may depend on the image content) and is potentially different for each element x.

4.2.2 Complexity Analysis

We recall that with a fast separable algorithm the EDT can be computed in $\mathcal{O}(|I|)$ time [24,40]. The Bernoulli sampling for each element using a pseudo-random number generator (PRNG) has complexity $\mathcal{O}(|\mathcal{Y}|)$, and the accumulation of each DT into a running sum (over the N sampling iterations) has complexity $\mathcal{O}(|I|)$. The complexity of the whole processes is therefore $\mathcal{O}(N|I|)$.

The method is highly parallelizable, since each random sampling and subsequent DT can be computed independently and are only aggregated at the end.

4.3 Deterministic Method

A deterministic estimation method for the SDT can be derived from an expansion of the expectation in Definition 1, observing an ordered sequence of distances to the nearest set points. For a given point in such a sequence, let its rank mean its index in the sequence.

Let $\mathcal{N}_{\mathcal{Y}}^{(k)}(x)$ denote $\mathcal{N}_{\text{supp}(\mathcal{Y})}^{(k)}(x)$, and $n_{\mathcal{Y}}^{(i)}(x)$ denote $n_{\text{supp}(\mathcal{Y})}^{(i)}(x)$, for convenience.

The probability that all nearest points in $\text{supp}(\mathcal{Y})$ of rank up to and including k do not belong to the set is the *avoidance functional* [41] which is given by

$$Q_{\mathcal{Y}}(\mathcal{N}_{\mathcal{Y}}^{(k)}(x)) = \prod_{i=1}^{k}\left[1 - p_{\mathcal{Y}}(n_{\mathcal{Y}}^{(i)}(x))\right]. \tag{17}$$

The probability corresponding to the i-th nearest point in $\text{supp}(\mathcal{Y})$ to be the nearest point to belong to the set (i.e., all the nearest points with lower rank than i do not belong to the set and the i-th nearest point does) is the *selection functional* which is given by

$$C_{\mathcal{Y}}(x;i) = \left\{ \begin{array}{ll} p_{\mathcal{Y}}(n_{\mathcal{Y}}^{(i)}(x)), & \text{for } i = 1 \\ p_{\mathcal{Y}}(n_{\mathcal{Y}}^{(i)}(x)) \cdot Q_{\mathcal{Y}}(\mathcal{N}_{\mathcal{Y}}^{(i-1)}(x)), & \text{for } 1 < i \leq |\mathcal{Y}| \\ 0, & \text{for } i > |\mathcal{Y}| \end{array} \right\}. \tag{18}$$

Definition 3 Given an image domain I, a DRS $\mathcal{Y}$, a maximal distance $d_{\texttt{MAX}} \in \mathbb{R}_+$, and a point-to-set distance d, a deterministic method for computing the SDT is

$$\begin{aligned} \texttt{DET-SDT}\left[\mathcal{Y}\right](x) = {} & Q_{\mathcal{Y}}(\mathcal{N}_{\mathcal{Y}}^{(|\mathcal{Y}|)}(x)) d_{\texttt{MAX}} \\ & + \sum_{i=1}^{|\mathcal{Y}|} C_{\mathcal{Y}}(x;i) \min\left[d(x, n_{\mathcal{Y}}^{(i)}(x)), d_{\texttt{MAX}}\right]. \end{aligned} \tag{19}$$

In practice, it may be computationally unfeasible or not required for the application at hand to compute the SDT exactly. The DET-SDT can, due to the diminishing weight of the high rank nearest points, be well approximated by truncating the distance probability distribution to either a given probability mass $p_{\texttt{MAX}}$ (in order of low-to-high rank) for the first k nearest points, thresholding all elements with probability below some threshold $p_{\min}$, or alternatively considering a fixed number k of nearest points. The k-nearest points and their distances can be computed using an efficient space-partitioning data structure, e.g., kd-trees [3] or by a separable algorithm [19].

We introduce some notions to aid the reasoning about the accumulated probability mass of sets of k-nearest neighbors. Let the accumulated probability mass up to and including the first i points be denoted as

$$p_{\text{acc}}\left[\mathcal{Y}, x\right](i) = \sum_{j=1}^{i} C_{\mathcal{Y}}(x;j). \tag{20}$$

For given $\mathcal{Y}$, x, $p_{\texttt{MAX}}$, let k_B be the smallest positive number such that

$$p_{\texttt{MAX}} \leq p_{\text{acc}}\left[\mathcal{Y}, x\right](k_B). \tag{21}$$

k_B encodes the smallest number of nearest neighbors which must be included for their accumulated probability to surpass or equal $p_{\texttt{MAX}}$. k_B will be used for computing lower and upper bounds on the SDT.

An upper bound on the SDT, where k_B is $k\left[\mathcal{Y}, x\right](p_{\texttt{MAX}})$, is

$$\begin{aligned} d_U\left[\mathcal{Y}\right](x) = {} & \sum_{i=1}^{k_B-1} C_{\mathcal{Y}}(x;i) \min\left[d(x, n_{\mathcal{Y}}^{(i)}(x)), d_{\texttt{MAX}}\right] \\ & + (p_{\texttt{MAX}} - p_{\text{acc}}\left[\mathcal{Y}, x\right](k_B - 1)) \min\left[d(x, n_{\mathcal{Y}}^{(k_B)}(x)), d_{\texttt{MAX}}\right] \\ & + (1 - p_{\texttt{MAX}}) d_{\texttt{MAX}}. \end{aligned} \tag{22}$$

The upper bound is based on truncating the tail of the distance distribution by substituting the distances of all higher rank (than k_B) nearest points with $d_{\texttt{MAX}}$. The k_B-th nearest point is only partially considered, in order to avoid spurious variations in distance across the DT due to variable probability mass being substituted with $d_{\texttt{MAX}}$. The upper bound is guaranteed to be tight if $p_{\texttt{MAX}} = 1$.

A lower bound on the SDT can be given by (assuming non-empty support and $p_{\texttt{MAX}} > 0$)

$$\begin{aligned} d_L^*\left[\mathcal{Y}\right](x) = {} & \sum_{i=1}^{k_B-1} C_{\mathcal{Y}}(x;i) \min\left[d(x, n_{\mathcal{Y}}^{(i)}(x)), d_{\texttt{MAX}}\right] \\ & + (p_{\texttt{MAX}} - p_{\text{acc}}\left[\mathcal{Y}, x\right](k_B - 1)) \min\left[d(x, n_{\mathcal{Y}}^{(k_B)}(x)), d_{\texttt{MAX}}\right], \end{aligned} \tag{23}$$

and based on d_L^*, a (tighter) normalized lower bound is given by

$$d_L\left[\mathcal{Y}\right](x) = \frac{1}{p_{\texttt{MAX}}} d_L^*\left[\mathcal{Y}\right](x). \tag{24}$$

The lower bound is based on truncating the tail of the distance distribution by substituting the distances of all higher rank (than k_B) nearest points (which have higher distances by definition) with the d_L^*, using normalization by division with $p_{\texttt{MAX}}$. The k_B-th nearest point is only partially considered, in order to avoid spurious variations in distance across the DT due to variable probability mass being substituted with d_L^*. The lower bound is guaranteed to be tight if $p_{\texttt{MAX}} = 1$ and $\text{core}(\mathcal{Y}) \neq \emptyset$.

Algorithm 1 computes both the upper bound and the lower bound jointly.

4.4 The Uniform Probability Model

If $p_{\mathcal{Y}}$ is of the form

$$p_{\mathcal{Y}}(x) = \left\{ \begin{array}{ll} \rho, & \text{for } x \in \hat{Y} \\ 0, & \text{for } x \notin \hat{Y} \end{array} \right\}, \tag{25}$$

given $\rho \in (0, 1]$, and input image $\hat{Y}$ then $p_{\mathcal{Y}}$ is a uniform CPF and we denote the corresponding DRS $\mathbb{U}[\hat{Y};\ \rho]$.

The uniform model is noteworthy, first and foremost for its simplicity, where a single probability value, together with an input binary image $\hat{Y}$ on domain I characterizes the entire DRS. Our initial study of the SDT [45] focused on this special case. Inserting Eq. (25) into Eq. (10) yields the distance transform

$$\texttt{SDT}\left[\mathbb{U}\left[\hat{Y};\ \rho\right]\right](x) = \mathbb{E}\left(\min\left[d\left(x, \mathbb{U}\left[\hat{Y};\ \rho\right]\right), d_{\texttt{MAX}}\right]\right), \quad \text{for } x \in I. \tag{26}$$

Algorithm 1 DET-SDT

Input: DRS $\mathcal{Y}$, domain I, max probability $p_{\mathtt{MAX}} \in (0, 1]$, and max distance $d_{\mathtt{MAX}} \in \mathcal{R}_+$.
Output: Lower and upper bound estimates (d_L, D_U) of $\mathtt{SDT}[\mathcal{Y}]$ for all points in I.

procedure DET- SDT- SP($\mathcal{Y}$, $p_{\mathtt{MAX}}$, $d_{\mathtt{MAX}}$, x)
  $d_{\text{acc}} \leftarrow 0$
  $q_0 \leftarrow 1$
  $p_{\text{acc}} \leftarrow 0$
  $i \leftarrow 1$
  while $i \leq |\mathcal{Y}|$ **and** $p_{\text{acc}} < p_{\mathtt{MAX}}$ **do**
    $c_i \leftarrow \min\left[p_{\mathcal{Y}}(n_{\mathcal{Y}}^{(i)}(x))q_{i-1}, p_{\mathtt{MAX}} - p_{\text{acc}}\right]$
    $d_{\text{acc}} \leftarrow d_{\text{acc}} + c_i \min\left[d(x, n_{\mathcal{Y}}^{(i)}(x)), d_{\mathtt{MAX}}\right]$
    $q_i \leftarrow (1 - p_{\mathcal{Y}}(n_{\mathcal{Y}}^{(i)}(x)))q_{(i-1)}$
    $p_{\text{acc}} \leftarrow p_{\text{acc}} + c_i$
    $i \leftarrow i + 1$
  end while
  return $(\frac{d_{\text{acc}}}{p_{\text{acc}}},\ d_{\text{acc}} + (1 - p_{\text{acc}})d_{\mathtt{MAX}})$
end procedure
procedure DET- SDT($\mathcal{Y}$, $p_{\mathtt{MAX}}$, $d_{\mathtt{MAX}}$)
  for $x \in I$ **do**
    $(d_{\mathrm{L}}(x), d_{\mathrm{U}}(x)) \leftarrow$ DET-SDT-SP($\mathcal{Y}$, $p_{\mathtt{MAX}}$, $d_{\mathtt{MAX}}$, x)
  end for
  return $(d_{\mathrm{L}}, d_{\mathrm{U}})$
end procedure

The uniform model provides a degree of robustness to additive (salt) noise, but treats all points in $\hat{Y}$ equally, which limits how much noise it can handle before corruption of the DT becomes severe. An illustration of the limitations of this model can be seen in Fig. 1h. Furthermore, it is ineffective in the presence of subtractive (pepper) noise, since only points in the input set $\hat{Y}$ are considered and missing points (points present in Y and absent in $\hat{Y}$) remain missing. Inside large connected components, the uniform model yields an overestimation of the distance with a bias of, at least,

$$\begin{aligned}\text{Bias}\left\{\mathtt{SDT}\left[\mathbb{U}\left[\hat{Y};\ \rho\right]\right]\right\}(x) &= (1-\rho)^{|\hat{Y}|}d_{\mathtt{MAX}} \\ &+ \sum_{i=2}^{|\hat{Y}|}\rho(1-\rho)^{i-1}\min\left[d(x, n_I^{(i)}(x)), d_{\mathtt{MAX}}\right],\end{aligned} \tag{27}$$

since the true distance is 0, and even if all object points are positioned as near to point x as possible, the ordered distance sequence of all the points in the domain from x will be included in the expectation with a nonzero probability.

The avoidance functional Q and selection functional C for the uniform model reduces to

$$Q_{\mathbb{U}\left[\hat{Y},\rho\right]}(\mathcal{N}_{\mathbb{U}\left[\hat{Y},\rho\right]}^{(i)}(x)) = (1-\rho)^i \tag{28}$$

and

$$C_{\mathbb{U}\left[\hat{Y},\rho\right]}(x; i) = \left\{\begin{array}{ll}\rho(1-\rho)^{i-1}, & \text{for } 1 \leq i \leq |\mathcal{Y}| \\ 0, & \text{for } i > |\mathcal{Y}|\end{array}\right\} \tag{29}$$

where $0^0 = 1$. The deterministic method therefore simplifies into

$$\begin{aligned}&\mathtt{DET\text{-}SDT}\left[\mathbb{U}\left[\hat{Y}, \rho\right]\right](x) = (1-\rho)^{|\hat{Y}|}d_{\mathtt{MAX}} + \\ &\sum_{i=1}^{|\hat{Y}|}\rho(1-\rho)^{i-1}\min\left[d(x, n_{\mathbb{U}\left[\hat{Y},\rho\right]}^{(i)}(x)), d_{\mathtt{MAX}}\right],\end{aligned} \tag{30}$$

in the uniform case, which is similar to the form of the deterministic formulation introduced in [45].

4.4.1 Approximation Error Analysis

The lower and upper bounds of the deterministic method both approach the true $\mathtt{SDT}[\mathcal{Y}](x)$ as $p_{\mathtt{MAX}} \to 1$. In this section, we investigate how fast the error decreases as a function of increasing $p_{\mathtt{MAX}}$, i.e., what rate of improvement is obtained when considering a larger part of the probability mass. We note that the magnitude of the error and the rate of improvement is highly image-dependent (and not constant over the image domain) but general error functions can be expressed in terms of the probabilities and distances.

For this error analysis, we use $p_{\text{acc}}(i)$ as shorthand for $p_{\text{acc}}[\mathcal{Y}, x](i)$.

If $p_{\mathtt{MAX}} < p_{\text{acc}}(|\mathcal{Y}|)$, then the error of the upper bound of the deterministic method as a function of the parameter $p_{\mathtt{MAX}}$ is

$$\begin{aligned}\epsilon_{d_U}[\mathcal{Y}, x](p_{\mathtt{MAX}}) &= (p_{\text{acc}}(k_B) - p_{\mathtt{MAX}})(d_{\mathtt{MAX}} - d(x, n_{\mathcal{Y}}^{(k_B)}(x))) \\ &+ \sum_{i=k_B+1}^{|\mathcal{Y}|} C_{\mathcal{Y}}(x;\ i)(d_{\mathtt{MAX}} - d(x, n_{\mathcal{Y}}^{(i)}(x))),\end{aligned} \tag{31}$$

and otherwise the error is zero. Only the term concerning point k_B varies continuously as a function of $p_{\mathtt{MAX}}$. If the error is not 0, the slope of the error function w.r.t. $p_{\mathtt{MAX}}$ is

$$\frac{d\epsilon_{d_U}[\mathcal{Y}, x]}{dp_{\mathtt{MAX}}}(p_{\mathtt{MAX}}) = d(x, n_{\mathcal{Y}}^{(k_B)}(x)) - d_{\mathtt{MAX}}, \tag{32}$$

which is non-positive since $d_{\mathtt{MAX}} \geq d_k$. The error function is thus a piece-wise linear non-increasing function where the difference between the maximum distance and the distance of point k yields the linear rate of improvement at a given $p_{\mathtt{MAX}}$.

For $p_{\text{MAX}} < p_{\text{acc}}(|\mathcal{Y}|)$, the (signed) error of the lower bound is given by a piece-wise reciprocal function

$$\begin{aligned}\epsilon_{d_L}[\mathcal{Y}, x](p_{\text{MAX}}) = &\frac{1}{p_{\text{MAX}}}\sum_{i=1}^{k_B-1} C_{\mathcal{Y}}(x;\, i) d(x, n_{\mathcal{Y}}^{(i)}(x)) \\ &+ \frac{1}{p_{\text{MAX}}}(p_{\text{MAX}} - p_{\text{acc}}(k_B - 1)) d(x, n_{\mathcal{Y}}^{(k_B)}(x)) \\ &- \text{SDT}[\mathcal{Y}](x),\end{aligned} \tag{33}$$

which has the derivative

$$\begin{aligned}\frac{d\epsilon_{d_L}[\mathcal{C}, x]}{dp_{\text{MAX}}}(p_{\text{MAX}}) = &\frac{1}{{p_{\text{MAX}}}^2}(p_{\text{acc}}(k_B - 1) d(x, n_{\mathcal{Y}}^{(k_B)}(x))) \\ &- \frac{1}{{p_{\text{MAX}}}^2}\sum_{i=1}^{k-1} C_{\mathcal{Y}}(x;\, i) d(x, n_{\mathcal{Y}}^{(i)}(x)).\end{aligned} \tag{34}$$

The derivative of the error of the lower bound can be interpreted as the change in error due to moving probability mass from the weighted sum of distances at p_{MAX} into the k-th point, weighted by the amount of probability mass already accumulated.

4.4.2 Complexity Analysis

Algorithm 1 is guaranteed to terminate, since I (and therefore also $\text{supp}(\mathcal{Y})$) is finite. It may also terminate early if the accumulated probability mass surpasses the chosen limit, p_{MAX}.

As was observed in [45] for the uniform model, an upper bound on the number of points k required to cover at least probability mass p_{MAX} for a DRS with minimum nonzero coverage probability $\min_+(\mathcal{Y})$ is given by

$$\kappa[\mathcal{Y}](p_{\text{MAX}}) = \left\lceil \frac{log(1 - p_{\text{MAX}})}{log(1 - \min_+(\mathcal{Y}))} \right\rceil. \tag{35}$$

The computational complexity of Algorithm 1 depends on the probabilities of the sequence of the k-nearest points, where k is unknown before executing the algorithm, and not constant over I. However, with the assumption that $\mathcal{Y}$ satisfies

$$\min_+(\mathcal{Y}) \geq \rho, \quad \text{for} \quad \rho \in [0,\, 1], \tag{36}$$

the complexity becomes $\mathcal{O}(k + D_{\kappa[\mathcal{Y}](p_{\text{MAX}})})$. In the worst case, $\mathcal{Y}(x) = \min_+(\mathcal{Y})$, $\forall x \in \text{supp}(\mathcal{Y})$. Computing the DET-SDT for all points in the domain I then has complexity $\mathcal{O}(|I|(k + D_k))$ where D_k denotes the computational complexity of finding the k nearest points.

The deterministic method is highly parallelizable, requiring only the prerequisite data structures (e.g., a kd-tree) to be initialized for fast computation of $n_{\mathcal{Y}}^{(i)}(x)$ and $d(x, n_{\mathcal{Y}}^{(i)}(x))$, since the evaluation of each point is an independent process.

5 Coverage Probability Estimation Using Kernel Density Estimation and Soft Thresholding

In this section, we introduce a method for constructing DRSs with non-uniform CPFs for the SDT based on density estimation or mass estimation. We propose using a variation of KDE for binary images on rectangular grids, to obtain a CPF based on the local density of object/foreground points.

Let $K : I \to [0, 1]$ be a discrete kernel, satisfying [48] (in the discrete case)

$$\sum_{x_1=-\infty}^{+\infty} \cdots \sum_{x_d=-\infty}^{+\infty} K(x_1, \ldots, x_d) = 1. \tag{37}$$

A DRS $\mathcal{Y}_K$ can then be formulated, which is characterized by a CPF given by the resulting kernel density estimate on an input set $\hat{Y}$,

$$p_{\mathcal{Y}_K}(x) = (F_{\hat{Y}} * K)(x), \tag{38}$$

where $*$ denotes the convolution operation which is used in the literature for fast KDE on discrete grids [51,52,60], and $F_{\hat{Y}}$ which denotes an indicator function of $\hat{Y}$. If a kernel K satisfies the normalization property (37) and has finite support, then each point x centered in a locally dense region, where all points of I falling within the kernel radius are object points in $\hat{Y}$, belongs to $\text{core}(\mathcal{Y}_K)$. If $x \in \text{core}(\mathcal{Y}_K)$, then $\text{SDT}[\mathcal{Y}_K](x) = 0$. The method therefore has zero bias within sufficiently large connected components.

The result of the KDE may not always have an ideal CPF for providing maximal accuracy of the DT. If objects of interest are small or thin, the maximal estimated density may be much lower than one, resulting in an empty core. In addition, sparse noise may not be fully suppressed, but kept to some degree, which can, in aggregate, have a large effect on the computed DT. We therefore propose to extend the KDE method with a soft thresholding operator, for images on rectangular grids (it could be generalized to other grids and to point-clouds by replacing the convolution with a point-cloud density estimation framework [9,63]). The soft thresholding operator we propose has parameterized crossover-point c, transition width w and steepness s, defined as

$$\tau_p[c, w, s](x) = \text{SMOOTH-STEP}_s\left(\frac{p(x) - c}{2w} + \frac{1}{2}\right), \tag{39}$$

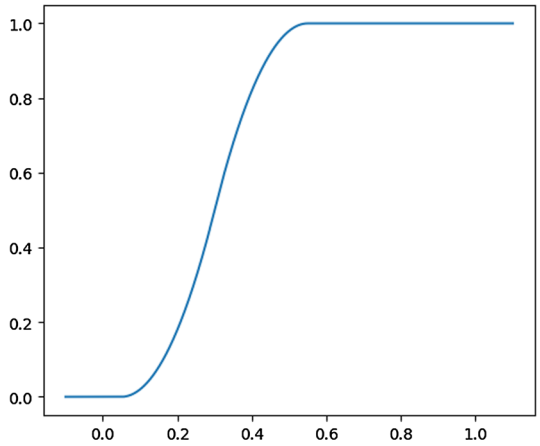


Fig. 2 The soft threshold function τ_p [0.3, 0.25, 2.0], (threshold 0.3, transition width 0.25, steepness 2)

where

$$\text{SMOOTH-STEP}_s(z) = \left\{ \begin{array}{ll} 0, & \text{for } z < 0 \\ 2^{s-1}(z^s), & \text{for } 0 \le z \le \frac{1}{2} \\ 1 - 2^{s-1}(1-z)^s, & \text{for } \frac{1}{2} < z \le 1 \\ 1, & \text{for } z > 1 \end{array} \right\}, \quad (40)$$

is a smooth transition function from 0 to 1. For $s = 1$, τ is a linear transition from 0 at $c - \frac{1}{2}w$, to 1 at $c + \frac{1}{2}w$; for $s \to \infty$, τ is a hard threshold at c. τ is flexible and expressive, and it has separate, continuous, and tunable parameters for the non-linear smoothing of the transition, the crossover point and the width transition. We observe that line structures benefit from a higher slope than disks. An example of the function τ, for a particular choice of the parameters, is given in Fig. 2. When the deterministic method for SDT computation is used, to reduce the potentially high computational cost of considering points with low CPs, we use a threshold $p_{\min}$, such that CPs below $p_{\min}$ are truncated to 0.

Inserting Eq. (38) into Eq. (39), we obtain, for a given discrete kernel K, a DRS with CPF

$$p\mathcal{Y}_{\hat{K}[c,w,s]}(x) = \tau_{p\mathcal{Y}_K}[c, w, s](x), \quad (41)$$

which we also shall denote $p\mathcal{Y}_{\hat{K}}$ when we omit the parameters, and the corresponding DRS $\mathcal{Y}_{\hat{K}}$ (Fig. 3).

If a kernel satisfies (37), it can be inserted into (38) and further into (41). This yields the distance transform

$$\text{SDT}\left[\mathcal{Y}_{\hat{K}}\right](x) = \mathbb{E}(\min\left[d(x, \mathcal{Y}_{\hat{K}}), d_{\text{MAX}}\right]), \quad \text{for } x \in I. \quad (42)$$

This method is denoted SDT_K in the performance analysis section.

Table 2 contains a summary of all the parameters used in the KDE CP estimation method, with their supported intervals.

6 Performance Analysis and Applications

In this section, we evaluate the proposed SDT using KDE-based CPs, as well as the model with uniform CPs, w.r.t. accuracy of the computed DT values. We also evaluate these methods on the task of segmentation of overlapping disks; we observe their performance when incorporating them in the watershed segmentation framework, which is closely related to, and relies on, DT. Since we observed similar performance for the Monte-Carlo and the deterministic method, [45], here we focus on the deterministic method, with an advantage of not including randomness. We present the evaluation results.

6.1 Alternative Methods for Computing Distance Transforms from Coverage Probability Functions

Here we describe two alternative methods which can be used together with the estimated CPF to compute DTs, based on hard thresholding and on distances to fuzzy sets. We present them here, to enable us to compare the proposed methods with alternative approaches.

If the parameter s in the function τ, Eq. (39), tends to ∞, and the SDT is replaced with the standard DT, we obtain a KDE-based denoising by thresholding method, which we denote T_K. This enables comparison between the SDT and KDE-based denoising.

Another class of tools which can utilize the known or estimated CPFs to compute distance transforms are distance measures from the fuzzy set-theory, [37,62]. The CPFs can be directly translated into membership values, which are defined for all elements and belong to the interval [0, 1], keeping in mind the difference in interpretation between probabilities and fuzzy membership values. The distance measure defined by Eq. (8) can thus be used to compute the distance from each point to the fuzzy set corresponding to $p\mathcal{Y}_K$ by assuming unit heights for all points of the domain, and integrating over the α-cuts,

$$d^\alpha(x, p\mathcal{Y}_{\hat{K}[c,w,s]}) = \int_0^1 \min\left[d(x, {}^\alpha p\mathcal{Y}_{\hat{K}[c,w,s]}), d_{\text{MAX}}\right] \text{d}\alpha. \quad (43)$$

We denote the fuzzy point-to-set method d_K^α.

6.2 Distance Accuracy in the Presence of Noise

To observe the accuracy of the proposed DTs compared to EDT and a few other reference methods, we evaluate them all on sets of synthetic generated images, with an aim to compare the obtained DTs with a ground-truth DT.

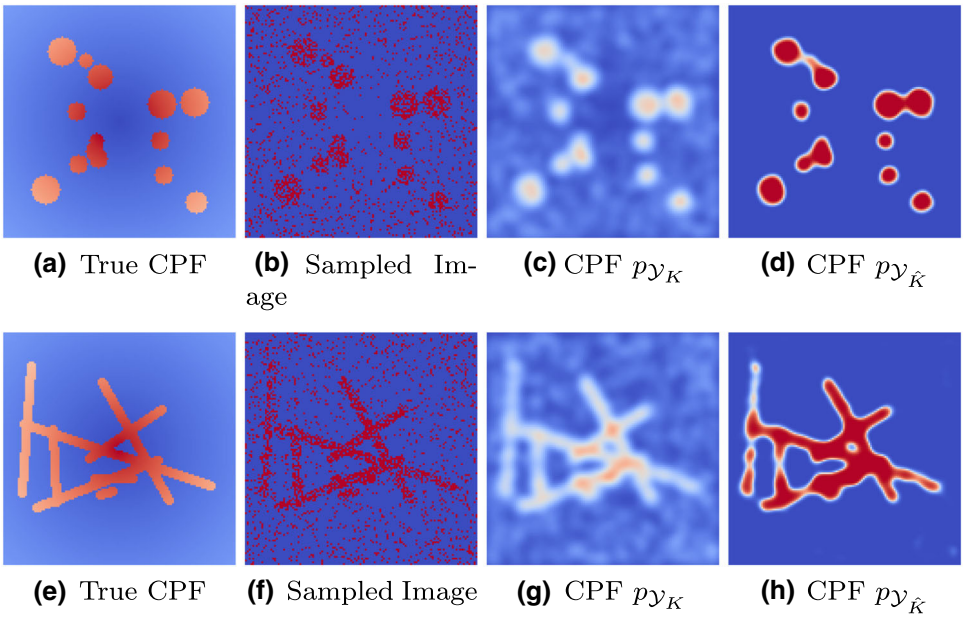


Fig. 3 **a**, **e** Visualization of two distributions which were sampled to obtain (**b**, **f**) noisy realizations. **c**, **g** CPF estimated from (**b**, **f**) using the K_{TW} kernel with radii 9 and 8.5 respectively. **d**, **h** CPF estimations based on **c**, **g** and soft thresholding $\mathcal{Y}_{\hat{K}[0.315,0.13,2.15]}$ and $\mathcal{Y}_{\hat{K}[0.315,0.24,4.35]}$ respectively

Table 2 All the parameters of the proposed method with the intervals of their allowed values

	Parameter description	Interval
r	Radius of kernel	$(0, \infty]$
c	Crossover point	$[0, 1]$
w	Transition width	$[0, 1]$
s	Steepness of the smooth step	$(0, \infty)$
$p_{\min}$	CP threshold (for DET-SDT)	$[0, 1]$

6.2.1 Synthetic Datasets

Two synthetic datasets of binary images of size 128×128 pixels were generated for evaluation of accuracy of the computed distance values: (i) a line segment image dataset, consisting of binary dilations of digital line segments drawn between random points and (ii) a disk image dataset, consisting of a random number (between 1 and 20) of digital disks, with random radii between 4 and 8. One version of each image is generated without noise, as a source of a reference distance transform, and one with between 1 to 20% additive noise and between 25 and 75% subtractive (pepper) noise, either increasing or decreasing as a function of the distances to the center-point of the image. For both datasets, we have created 100 (pairs of) training images, for parameter-tuning, and 1000 (pairs of) test images. Figure 4 shows examples from both of the datasets, with and without corruption by noise.

6.2.2 Experiment Setup

Let D be a dataset of N distance transforms generated by some method and G be a dataset of corresponding N ground-truth distance transforms, on domain I. The used quality measures for the error of the distance transforms over datasets D and G are the mean average error (MAE) of the distance values, given by

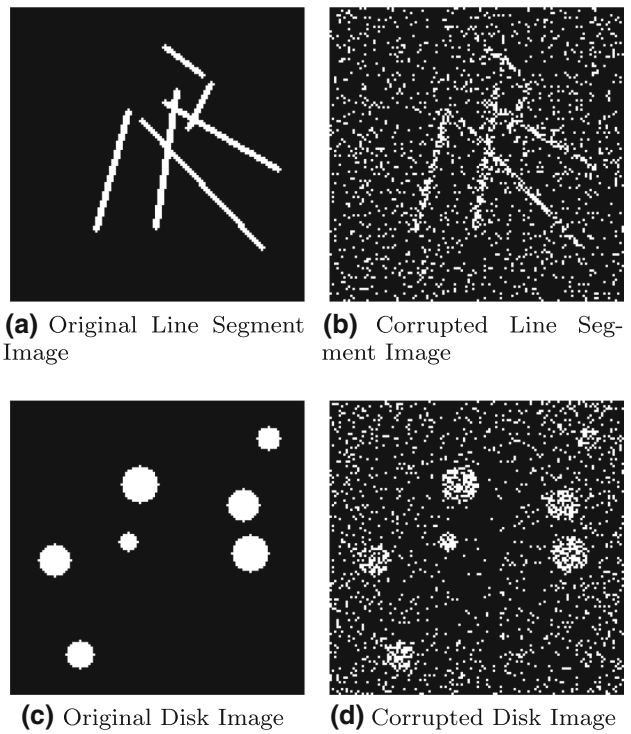


Fig. 4 Example images from the line segment image dataset and the disk image dataset, showing the original images (**a**, **c**) and corrupted versions (**b**, **d**)

$$\mathrm{MAE}_{D,G} = \frac{1}{N}\frac{1}{|I|}\sum_{i=1}^{N}\sum_{x\in I}|D_i(x) - G_i(x)|, \tag{44}$$

and the mean maximum error (MME) of the distance values, defined as

$$\mathrm{MME}_{D,G} = \frac{1}{N}\sum_{i=1}^{N}\max_{x\in I}|D_i(x) - G_i(x)|, \tag{45}$$

where D_i is the estimated distance transform for image i in the dataset D and G_i is the ground-truth distance transform for the noise-free image i in the dataset G. In addition to these mean error measures, the empirical standard deviations are also presented.

For both datasets and the deterministic SDT method, a probability mass of $p_{\mathrm{MAX}} = 0.95$ and a truncation probability threshold $p_{\min} = 0.05$ were used. The presented results use the lower bound estimates from Algorithm 1.

6.2.3 Parameter Tuning

The parameters, listed in Table 2 (with the exception of $p_{\min}$), were tuned by 600 iterations of random search in the parameter space on the training sets (with 100 images) for both datasets, within ranges given in Table 3. Along with the tuning of the SDT method, the hard-thresholding method and the fuzzy set distance method were also tuned, and the best parameter for each method was chosen. Normalization of the CPF, performed according to

$$\tilde{p}_{\mathcal{Y}}(x) = \frac{p_{\mathcal{Y}}(x) - \min_{x\in I} p_{\mathcal{Y}}}{\max_{x\in I} p_{\mathcal{Y}} - \min_{x\in I} p_{\mathcal{Y}}}, \tag{46}$$

for KDE-based DRS $\mathcal{Y}_K$, is applied before the thresholding step to investigate if it has beneficial effects on the accuracy of the DT. The parameters of each method, for which the lowest MAE was observed on the training set, were then used for the evaluation on the test sets.

For the line segment data set, the parameters leading to best performance are: $r = 8.1, c = 0.15, w = 0.29, s = 4.65$. For the disk data set, the optimal parameter values are: $r = 9, c = 0.315, w = 0.13, s = 2.15$. We observe a slight difference in the transition range and the steepness parameters, but otherwise very similar values are identified. Normalization was not beneficial on either of the datasets for the SDT-method, but it did improve the accuracy of the hard thresholding.

6.2.4 Choice of Kernel

With the aim of choosing a kernel which maximizes the accuracy of the final SDT, a number of commonly used kernels were considered. To evaluate the kernels, each one was tuned (with 600 iterations of random search) on the combined disk and line segment datasets, containing 200 training images and evaluated on the 2000 test images in the combined set. We list the MAEs from this evaluation in Table 4. The Triweight kernel, K_{TW}, exhibited the highest accuracy and was therefore chosen as the kernel candidate for the rest of the experiments. The same kernel, K_{TW}, led to the best observed performance of T_K and d_K^α, as well.

Table 3 Parameters included in parameter tuning, and their ranges

Param.	Values
r	$\{8.0, 8.1, \ldots, 11.9, 12.0\}$
s	$\{0.25, 0.35, \ldots, 4.85, 4.95\}$
c	$\{0.300, 0.305, \ldots, 0.495, 0.500\}$
w	$\{0.01, 0.02, \ldots, 0.49, 0.50\}$
Normalization of KDE	{Off, On}

Table 4 Results of the evaluation of different kernels in terms of mean (over different images) average (over different pixels within each image) error

Kernel	Mean avg. err.
K_{U}	5.666 ± 8.288
K_{T}	2.952 ± 3.975
K_{BW}	2.781 ± 3.804
K_{TW}	$\mathbf{2.580 \pm 3.479}$
K_{E}	3.246 ± 4.148
$K_{\cos}$	3.207 ± 4.131

The kernel exhibiting the highest accuracy is marked in bold

6.2.5 Results

The results of the evaluation are presented in Table 5. The proposed method (SDT_K) substantially outperforms the reference methods (EDT, EDT+Opening, k-distance, T_K and d_K^α) on both datasets and both considered quality measures.

6.3 Algorithm Convergence and Run-time Analysis

In this work, we propose two algorithms for computing the SDT. To assist in the choice of algorithm, the empirical rate of convergence to the exact SDT (which can be computed exactly with the deterministic algorithm if $p_{\mathrm{MAX}} = 1$ and the upper bound is used) and run-time are measured and contrasted.

6.3.1 Experiment Setup

Starting from an image from the line segment data set, used in the synthetic tests of Sect. 6.2, we compute the exact SDT and then compute distance and elapsed run-time measured in

Table 5 Accuracy of the considered methods on the synthetic datasets as measured in mean average error (MAE) and mean maximum error (MME)

Method	Line segment data set		Disk data set	
	Mean Avg. Err.	Mean max. err.	Mean avg. err.	Mean max. err.
EDT	17.55 ± 7.64	53.61 ± 12.99	18.75 ± 10.46	63.77 ± 21.81
EDT+Opening	9.99 ± 12.38	38.28 ± 25.60	7.89 ± 8.67	27.84 ± 20.87
k-distance	10.49 ± 5.19	34.38 ± 11.60	11.07 ± 6.03	31.72 ± 16.85
T_K	6.41 ± 6.96	27.63 ± 20.53	2.51 ± 6.12	11.18 ± 13.68
d_K^α	8.23 ± 4.01	26.32 ± 9.04	7.19 ± 5.56	22.98 ± 11.86
SDT_U	13.44 ± 7.06	45.25 ± 12.23	14.63 ± 9.82	53.25 ± 22.20
SDT_K	$\mathbf{2.97 \pm 2.76}$	$\mathbf{16.96 \pm 12.59}$	$\mathbf{2.01 \pm 5.40}$	$\mathbf{10.00 \pm 12.80}$

Lower is better
Bold values indicate best result out of the considered methods

seconds, for a sequence of p_{MAX} and N-values, respectively. We then use the mean average error (MAE) evaluation criterion between the exact SDT and each approximation to determine the accuracy.

The parameters chosen for this experiment are $r = 4.0, c = 0.15, w = 0.3, s = 1.0, p_{\min} = 0.05$. The kernel is chosen to be K_{TW}. These parameters are not meant to be optimal in terms of accuracy for the test image but provide a wide range of different probability values in the resulting CPF in order to exercise the algorithms.

The research prototype implementation of the deterministic algorithm used for this experiment uses a kd-tree data structure provided in the SciPy software package [59], which does not take advantage of the regular grid structure of the data. This is a limitation of the part of this study related to measurement of run-time.

All experiments are performed on a workstation with a 6-core Intel i7-6800K, 3.4 GHz processor with 48GB of RAM and 15 MB cache. The operating system Ubuntu 18.04 LTS and Python version 3.6.9 were used.

6.3.2 Results

In Fig. 5, the MAE and run-time for each method are displayed as functions of the respective method's key approximation parameter. For the deterministic method, the MAE decreases approximately linearly as a function of p_{MAX}, while the run-time increases super-linearly. For the Monte Carlo algorithm, the MAE decreases approximately as predicted by Eq. (16), which was evaluated by observing a sequence $N = \{100, \ldots, 250\}$ and computing the quotients of the MAE for data-point N and MAE for data-point $4N$. The mean of these quotients measured 1.992 which is close to the expected 2.

If the requirements of accuracy are not of highest importance and non-deterministic output is not a concern, the Monte Carlo algorithm with a low N is a good choice. If very high accuracy is required, or if non-deterministic distance values are not suitable for the application at hand, the deterministic algorithm is the clear choice.

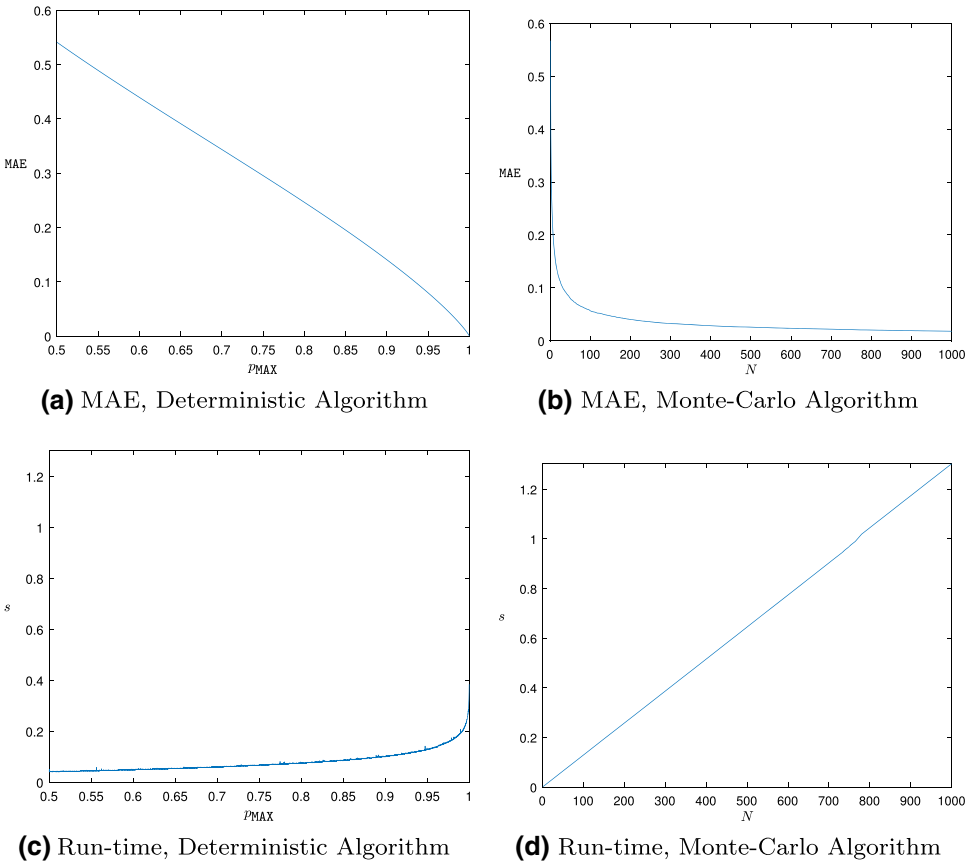
6.4 Watershed Segmentation of Overlapping Objects

Watershed segmentation [5] is a prominent method for segmentation tasks where touching or overlapping objects are to be separated (assigned distinct labels). It is an important part of many pipelines for, e.g., cell segmentation and tracking [11]. More recent approaches combine convolutional neural networks with watershed partitioning [61]. One challenge with using the classical watershed method is that it is prone to over-segmentation, where parts of a single object are split into multiple objects, necessitating post-processing to fuse those objects back together. The over-segmentation can be caused by spurious local maxima (caused by irregularities and noise at the object boundaries) in the DT (to the background) used to compute the watershed. In this section, we investigate if the SDT, being smooth and robust to noise, can reduce the impact of noisy object boundaries. This evaluation is an extension to the performance analysis of [45].

6.4.1 Experiment Setup

We consider a set of distances $\{3.0, 3.1, \ldots, 9.8, 9.9\}$ at which we position two disks with radius in the range $[\pi, 2\pi]$ in a digital grid, where the centroid of one of the disks is positioned at a random sub-pixel position and the centroid of the other disk is placed at the given distance from the first disk, displaced in a random direction. For each distance, 500 test cases are generated and segmented with the considered methods. The evaluation is performed both without any noise (in addition to the discretization artifacts) and with various amounts of salt and pepper noise applied at the boundaries of the objects, which serves to create random deformations (cavities and protrusions).

A simplified scenario with only two objects in the image is considered because of two main reasons: (i) it simplifies the



(a) MAE, Deterministic Algorithm

(b) MAE, Monte-Carlo Algorithm

(c) Run-time, Deterministic Algorithm

(d) Run-time, Monte-Carlo Algorithm

Fig. 5 The MAE error and run-time of the two proposed SDT algorithms as functions of their respective parameters. Trend lines added for illustration purposes

evaluation procedure and (ii) even in fully realistic scenarios with overlapping circular objects, they would not all be overlapping but instead consist of a large number of connected components each being treated individually by the watershed segmentation.

For the standard EDT, we also consider a version where a morphological closing (dilation followed by erosion) with a 3×3 square structuring element is applied before computing the DT, to eliminate some of the noise.

Success is defined as segmenting exactly two objects and where the centroids of the two true objects are assigned distinct labels.

6.4.2 Parameter Tuning

The parameters for the different methods were tuned using 25 images for each distance. For the SDT_U method, a coverage probability value $\rho = 0.25$ was chosen, and $p_{\mathrm{MAX}} = 0.75$. For the SDT_K method, transition size $w = 0.5$, crossover point $c = 0.35$, radius $r = 2.5$, smooth-step exponent $s = 0.5$, and normalization disabled, and $p_{\mathrm{MAX}} = 0.9$. For the SDT_K method, it increased performance to mask out (assign CP to 0) all points which were not included in the original noisy image. For the T_K method, threshold $c = 0.1$, radius $r = 2.5$, normalization disabled. For the d_K^α, all the relevant parameters had the same values as for SDT_K, and 15 quantization levels were used. For the k-distance, $k = 6$.

6.4.3 Results

The results of the watershed segmentation experiments are presented in Fig. 6 and Table 6. We observe that the SDT-based methods are superior to the other compared methods for the task of interest and that both the uniform and the KDE-based SDTs exhibit similar performance, with a slight advantage to SDT_K for all cases.

6.5 Distance Transforms of Images in Localization Microscopy

Stochastic Optical Reconstruction Microscopy (STORM) [4] is a family of imaging methods which enables imaging nanometer-scale structures with a very fine spatial resolution. By imaging only a small subset of the objects of interest in every image taken in a series, the locations of the objects can be computed by fitting a point-spread function (PSF) model over the modes of each image. Each image thus contributes a sparse set of points for a final point-cloud constituting the detected object points.

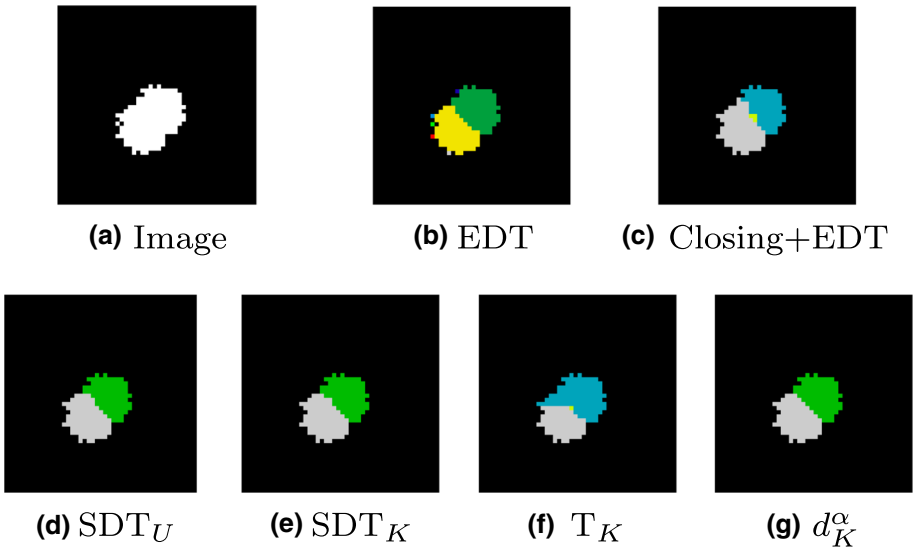


Fig. 6 Example of the result of watershed segmentation of an image using the considered methods

Table 6 Success rate of the watershed segmentation of overlapping (circular) objects subject to random cavities and protrusions for a number of chosen methods

Method	SR (noise free)	SR (noise: 0.1)	SR (noise: 0.25)	SR (noise: 0.5)
EDT	0.552	0.146	0.043	0.046
EDT+Closing	0.554	0.466	0.394	0.339
k-distance	0.708	0.677	0.638	0.603
T_K	0.367	0.365	0.354	0.345
d_K^α	0.646	0.631	0.602	0.570
SDT_U	0.716	0.686	0.650	0.617
SDT_K	**0.718**	**0.692**	**0.659**	**0.621**

Bold values indicate best result out of the considered methods

6.5.1 Experimental Setup

We used the TestSTORM simulator [43,53] for generating test image series for a known synthetic test pattern consisting of over 30,000 points in a point cloud forming a set of lines. This is a synthetic but realistic test scenario since one major use-case of this type of microscopy is to image thin curve structures [53]. Using simulated data enables the computation of a distortion-free ground-truth image and corresponding distance transform, enabling direct evaluation. We simulate 5 series of 800 different 64×64 images with short exposure time (0.005 s) and pixel size 160 nanometers and then compute reconstructed images from the simulated image series using the software RainSTORM [43]. We then create higher resolution images of size 256×256 from the resulting point-clouds. All the considered methods are then used to compute DTs to these noisy images, and the results are compared with the ground-truth DT to obtain mean average error (MAE) and mean maximum error (MME) in nanometers.

6.5.2 Parameter Tuning

We generated a set of 5 training images, with the identical process as for the test images. For each DT method which has parameters to set, we generated 1000 random sets of parameters, and the method was subsequently applied to all the training images and the set of parameters which resulted in the lowest mean average error (MAE) was chosen for processing of the test set.

6.5.3 Results

The quantitative results are presented in Table 7. We observe that the results for this (more realistic) set of images agree to a high extent in terms of accuracy with the results on the line segment image dataset; the SDT_K method has a MAE which is less than half the MAE of the second best evaluated method. An example of the spatial distribution of the error over the image, as well as magnitude of the error, are shown in Fig. 7.

Table 7 Accuracy of the considered methods on the STORM microscopy simulated datasets as measured in mean average error (MAE) [nanometers] and mean maximum error (MME) [nanometers]

Method	Training set		Test set	
	Mean avg. err.	Mean max. err.	Mean avg. err.	Mean max. err.
EDT	1008.75	2770.41	1008.78	2752.09
EDT+Opening	128.17	876.64	144.02	939.26
k-distance	380.86	1214.12	386.67	1269.56
T_K	122.07	698.06	144.03	932.30
d_K^α	318.24	1125.92	305.45	1153.80
SDT_U	632.78	2044.60	636.00	2049.57
SDT_K	**62.68**	**384.97**	**68.93**	**476.79**

Lower is better
Bold values indicate best result out of the considered methods

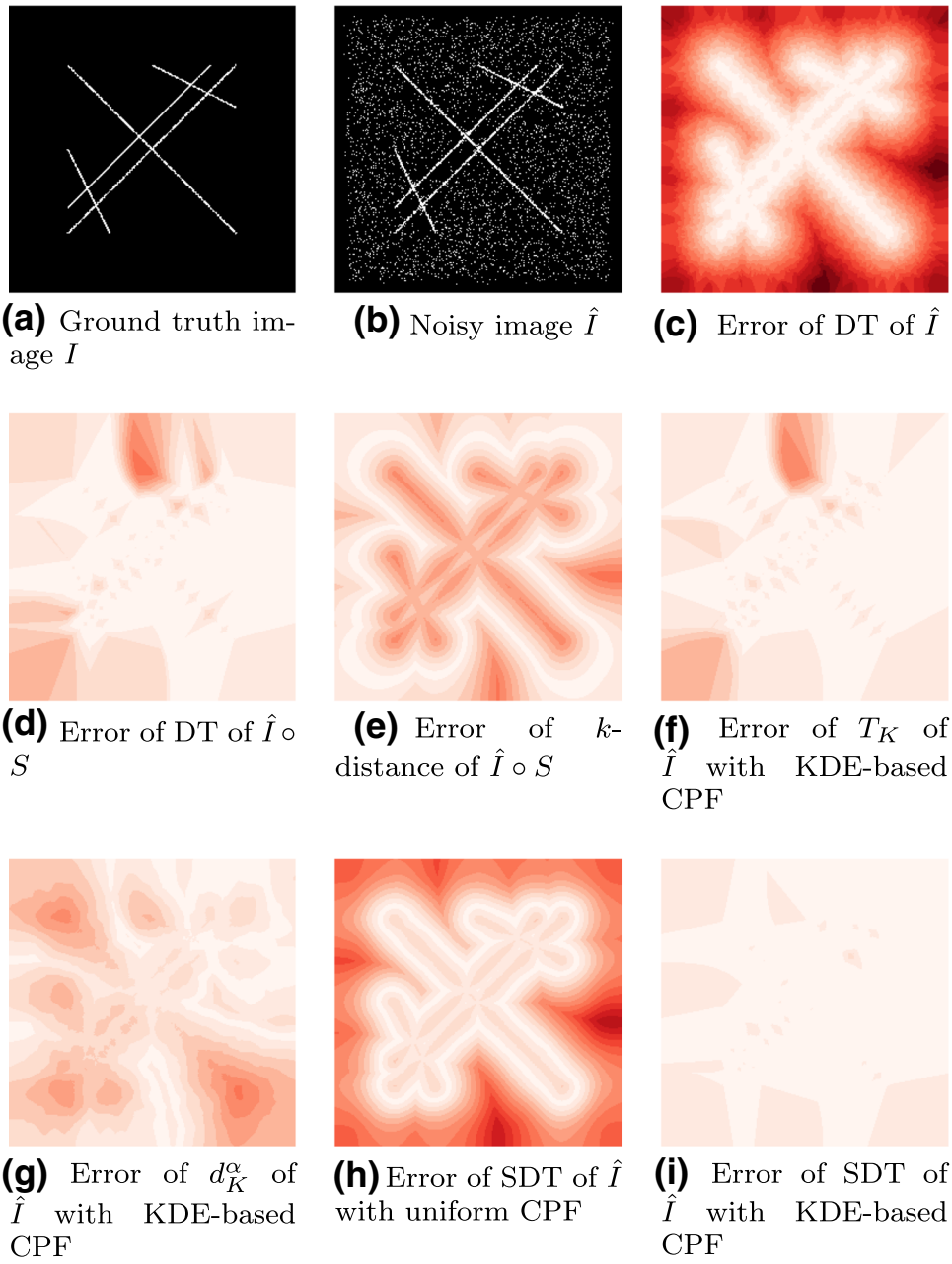


Fig. 7 Visualization of the error of different methods on DT on point-clouds yielded by STORM imaging. **a** Ground-truth image; **b** an acquired image from **a** corrupted by noise resulting from imaging artifacts and spurious signal detections; **c–i** the errors of the considered methods, computed with **b** as input, compared to the DT of the ground-truth image. All color-maps are the same for **c–i** with discretized colors representing different errors. Red signifies higher error (Color figure online)

7 Conclusion and Future Work

In this work, we present the SDT, a DT which uses a model of noisy images based on DRSs to achieve robustness in the presence of noise, and develop it for general DRS (satisfying mutual independence of inclusion of elements). The work extends the definition introduced in [45] (where a preliminary study of SDT, observed on a special case of a uniform CPs, is presented). We propose a density-based method for estimating the CPs, which is able to separate noise and objects well, without relying on, and thus being sensitive to exact local patterns, which can be a challenge for, e.g., morphological noise filtering. We show through empirical evaluation that excellent accuracy of the proposed SDT, relying on the KDE-based CP estimation method, can be attained using the proposed method, even in the presence of severe corruption (1–20% of additive noise and between 25 and 75% of subtractive noise) of the image of interest, substantially better than the other compared methods (both hard-thresholding of KDE, a method based on fuzzy set theory, the SDT with a uniform CPF, morphological opening-based denoising followed by a standard DT, and the k-distance [18]).

We also show that the method can be successfully used together with watershed segmentation for separation of overlapping objects with increased robustness to both discretization artifacts and noise appearing on the border of the objects, which have a strong negative effect on segmentation which relies on a classic (noise-sensitive) DT.

We further evaluate the method on simulated STORM localization microscopy images where we observed very good results, congruent with the larger synthetic study.

For future work along this research track, we suggest exploring learning-based methods for estimation of CPFs, using, e.g., denoising auto-encoders, which have capacities to model more complex patterns than a single kernel, faster specialized kd-tree algorithms for accelerating the computation of the SDT on rectangular grids, and exploring the applicability and performance of the SDT in image registration.

Acknowledgements Open access funding provided by Uppsala University.

Compliance with Ethical Standards

Conflict of interest The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Baddeley, A., Molchanov, I.: Averaging of random sets based on their distance functions. J. Math. Imaging Vis. **8**(1), 79–92 (1998)
2. Benamou, J.D., Carlier, G., Cuturi, M., Nenna, L., Peyré, G.: Iterative Bregman projections for regularized transportation problems. SIAM J. Sci. Comput. **37**(2), A1111–A1138 (2015)
3. Bentley, J.L.: Multidimensional binary search trees used for associative searching. Commun. ACM **18**(9), 509–517 (1975)
4. Betzig, E., Patterson, G.H., Sougrat, R., Lindwasser, O.W., Olenych, S., Bonifacino, J.S., Davidson, M.W., Lippincott-Schwartz, J., Hess, H.F.: Imaging intracellular fluorescent proteins at nanometer resolution. Science **313**(5793), 1642–1645 (2006)
5. Beucher, S.: Use of watersheds in contour detection. In: Proceedings of the international workshop on image processing. CCETT (1979)
6. Borgefors, G.: Distance transformations in arbitrary dimensions. Comput. Vis. Gr. Image Process. **27**(3), 321–345 (1984)
7. Borgefors, G.: Distance transformations in digital images. Comput. Vis. Gr. Image Process. **34**(3), 344–371 (1986)
8. Borgefors, G.: Hierarchical chamfer matching: a parametric edge matching algorithm. IEEE Trans. Pattern Anal. Mach. Intell. **10**(6), 849–865 (1988)
9. Brophy, M., Beauchemin, S., Barron, J.: Kernel density filtering for noisy point clouds in one step. In: Irish Machine Vision and Image Processing Conference Proceedings (2015)
10. Bubenik, P., Hull, M., Patel, D., Whittle, B.: Persistent homology detects curvature. Inverse Probl. **36**(2), 025008 (2019)
11. Carpenter, A.E., Jones, T.R., Lamprecht, M.R., Clarke, C., Kang, I.H., Friman, O., Guertin, D.A., Chang, J.H., Lindquist, R.A., Moffat, J., et al.: Cellprofiler: image analysis software for identifying and quantifying cell phenotypes. Genome Biol. **7**(10), R100 (2006)
12. Chazal, F., Cohen-Steiner, D., Mérigot, Q.: Geometric inference for probability measures. Found. Comput. Math. **11**(6), 733–751 (2011)
13. Chazal, F., Fasy, B., Lecci, F., Michel, B., Rinaldo, A., Rinaldo, A., Wasserman, L.: Robust topological inference: distance to a measure and kernel distance. J. Mach. Learn. Res. **18**(1), 5845–5884 (2017)
14. Chintakunta, H., Gentimis, T., Gonzalez-Diaz, R., Jimenez, M.J., Krim, H.: An entropy-based persistence barcode. Pattern Recogn. **48**(2), 391–401 (2015)
15. Coeurjolly, D.: 2D subquadratic separable distance transformation for path-based norms. In: International Conference on Discrete Geometry for Computer Imagery, pp. 75–87. Springer (2014)
16. Coeurjolly, D., Vacavant, A.: Separable distance transformation and its applications. In: Brimkov, V.E., Barneva, R.P. (eds.) Digital Geometry Algorithms, pp. 189–214. Springer, Berlin (2012)
17. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. **24**(5), 603–619 (2002)
18. Cuel, L., Lachaud, J.O., Mérigot, Q., Thibert, B.: Robust geometry estimation using the generalized Voronoi covariance measure. SIAM J. Imaging Sci. **8**(2), 1293–1314 (2015)

19. Cuisenaire, O., Macq, B.: Fast k-NN classification with an optimal k-distance transformation algorithm. In: 2000 10th European on Signal Processing Conference, pp. 1–4. IEEE (2000)
20. Ćurić, V., Lindblad, J., Sladoje, N., Sarve, H., Borgefors, G.: A new set distance and its application to shape registration. Pattern Anal. Appl. **17**(1), 141–152 (2014)
21. Cuturi, M.: Sinkhorn distances: Lightspeed computation of optimal transport. In: Advances in neural information processing systems, pp. 2292–2300 (2013)
22. Edelsbrunner, H., Harer, J.: Persistent homology: A survey. Contemp. Math. **453**, 257–282 (2008)
23. Epanechnikov, V.A.: Non-parametric estimation of a multivariate probability density. Theory Probab. Appl. **14**(1), 153–158 (1969)
24. Felzenszwalb, P.F., Huttenlocher, D.P.: Distance transforms of sampled functions. Theory Comput. **8**(1), 415–428 (2012)
25. Goutsias, J.: Morphological analysis of discrete random shapes. J. Math. Imaging Vis. **2**(2–3), 193–215 (1992)
26. Goutsias, J.I., Wen, C.: Discrete random set models for shape synthesis and analysis. In: Visual Communications and Image Processing'91: Image Processing, vol. 1606, pp. 174–186. International Society for Optics and Photonics (1991)
27. Guibas, L., Morozov, D., Mérigot, Q.: Witnessed k-distance. Discrete Comput. Geom. **49**(1), 22–45 (2013)
28. Gustavson, S., Strand, R.: Anti-aliased Euclidean distance transform. Pattern Recogn. Lett. **32**(2), 252–257 (2011)
29. Ikonen, L., Toivanen, P.: Distance and nearest neighbor transforms on gray-level surfaces. Pattern Recogn. Lett. **28**(5), 604–612 (2007)
30. Ilić, V., Lindblad, J., Sladoje, N.: Precise Euclidean distance transforms in 3D from voxel coverage representation. Pattern Recogn. Lett. **65**, 184–191 (2015)
31. Jones, M.C.: The performance of kernel density functions in kernel distribution function estimation. Stat. Probab. Lett. **9**(2), 129–132 (1990)
32. Langrené, N., Warin, X.: Fast and stable multivariate kernel density estimation by fast sum updating. J. Comput. Gr. Stat. **28**, 596–608 (2019)
33. Latecki, L.J., Lazarevic, A., Pokrajac, D.: Outlier detection with kernel density functions. In: International Workshop on Machine Learning and Data Mining in Pattern Recognition, pp. 61–75. Springer (2007)
34. Levi, G., Montanari, U.: A grey-weighted skeleton. Inf. Control **17**(1), 62–91 (1970)
35. Lewis, T., Owens, R., Baddeley, A.: Averaging feature maps. Pattern Recogn. **32**(9), 1615–1630 (1999)
36. Lindblad, J., Bengtsson, E.: A comparison of methods for estimation of intensity non uniformities in 2d and 3d microscope images of fluorescence stained cells. In: Proceedings of the Scandinavian Conference on Image Analysis, pp. 264–271 (2001)
37. Lindblad, J., Sladoje, N.: Linear time distances between fuzzy sets with applications to pattern matching and classification. IEEE Trans. Image Process. **23**(1), 126–136 (2014)
38. Lindblad, J., Sladoje, N.: Exact linear time Euclidean distance transforms of grid line sampled shapes. In: International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing, pp. 645–656. Springer (2015)
39. Matheron, G.: Random Sets and Integral Geometry. Wiley, New York (1975)
40. Maurer, C.R., 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**(2), 265–270 (2003)
41. Molchanov, I., Molchanov, I.S.: Theory of Random Sets, vol. 19. Springer, Berlin (2005)
42. Moreno, R., Koppal, S., De Muinck, E.: Robust estimation of distance between sets of points. Pattern Recogn. Lett. **34**(16), 2192–2198 (2013)
43. Novák, T., Gajdos, T., Sinkó, J., Szabó, G., Erdélyi, M.: Teststorm: versatile simulator software for multimodal super-resolution localization fluorescence microscopy. Sci. Rep. **7**(1), 1–8 (2017)
44. Öfverstedt, J., Lindblad, J., Sladoje, N.: Fast and robust symmetric image registration based on distances combining intensity and spatial information. IEEE Trans. Image Process. **28**(7), 3584–3597 (2019)
45. Öfverstedt, J., Lindblad, J., Sladoje, N.: Stochastic distance transform. In: International Conference on Discrete Geometry for Computer Imagery, pp. 75–86. Springer (2019)
46. Rosenfeld, A., Pfaltz, J.L.: Sequential operations in digital picture processing. J. ACM (JACM) **13**(4), 471–494 (1966)
47. Rosenfeld, A., Pfaltz, J.L.: Distance functions on digital pictures. Pattern Recogn. **1**(1), 33–61 (1968)
48. Rossi, V., Vila, J.P.: Nonlinear filtering in discrete time: a particle convolution approach. Annales de l'Institut de Statistique de l'Université de Paris **50**(3), 71–102 (2006)
49. Saha, P.K., Borgefors, G., di Baja, G.S.: A survey on skeletonization algorithms and their applications. Pattern Recogn. Lett. **76**, 3–12 (2016)
50. Saha, P.K., Wehrli, F.W., Gomberg, B.R.: Fuzzy distance transform: theory, algorithms, and applications. Comput. Vis. Image Underst. **86**(3), 171–190 (2002)
51. Schubert, E., Zimek, A., Kriegel, H.P.: Generalized outlier detection with flexible kernel density estimates. In: Proceedings of the 2014 SIAM International Conference on Data Mining, pp. 542–550. SIAM (2014)
52. Silverman, B.W.: Algorithm as 176: kernel density estimation using the fast Fourier transform. J. R. Stat. Soc. Ser. C (Appl. Stat.) **31**(1), 93–99 (1982)
53. Sinkó, J., Kákonyi, R., Rees, E., Metcalf, D., Knight, A.E., Kaminski, C.F., Szabó, G., Erdélyi, M.: Teststorm: simulator for optimizing sample labeling and image acquisition in localization based super-resolution microscopy. Biomed. Opt. Express **5**(3), 778–787 (2014)
54. Sivignon, I.: Representation of imprecise digital objects. In: International Conference on Discrete Geometry for Computer Imagery, pp. 401–414. Springer (2016)
55. Solomon, J., De Goes, F., Peyré, G., Cuturi, M., Butscher, A., Nguyen, A., Du, T., Guibas, L.: Convolutional Wasserstein distances: efficient optimal transportation on geometric domains. ACM Trans. Gr. (TOG) **34**(4), 1–11 (2015)
56. Vacavant, A.: Fast distance transformation on irregular two-dimensional grids. Pattern Recogn. **43**(10), 3348–3358 (2010)
57. Vacavant, A., Coeurjolly, D., Tougne, L.: Distance transformation on two-dimensional irregular isothetic grids. In: International Conference on Discrete Geometry for Computer Imagery, pp. 238–249. Springer (2008)
58. Vacavant, A., Coeurjolly, D., Tougne, L.: Separable algorithms for distance transformations on irregular grids. Pattern Recogn. Lett. **32**(9), 1356–1364 (2011)
59. Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., et al.: Scipy 1.0: fundamental algorithms for scientific computing in python. Nat. Methods **17**(3), 261–272 (2020)
60. Wand, M.: Fast computation of multivariate kernel estimators. J. Comput. Gr. Stat. **3**(4), 433–445 (1994)
61. Wolf, S., Pape, C., Bailoni, A., Rahaman, N., Kreshuk, A., Kothe, U., Hamprecht, F.: The mutex watershed: efficient, parameter-free image partitioning. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 546–562 (2018)
62. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**(3), 338–353 (1965)
63. Zaman, F., Wong, Y.P., Ng, B.Y.: Density-based denoising of point cloud. In: 9th International Conference on Robotic, Vision, Signal Processing and Power Applications, pp. 287–295. Springer (2017)

64. Zomorodian, A., Carlsson, G.: Computing persistent homology. Discrete Comput. Geom. **33**(2), 249–274 (2005)

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

Johan Öfverstedt received the B.Sc. and M.Sc. degrees in Computer Science in 2012 and 2017 respectively, and is currently pursuing a Ph.D. degree in Computerized Image Processing at Centre for Image Analysis, Uppsala University, Sweden, since 2017. His research interests include distance measures, image registration, object recognition, and optimization.

Joakim Lindblad received the M.Sc. in Engineering Physics and Ph.D. in Computerized Image Analysis from Uppsala University, Sweden, in 1997 and 2003, respectively. He is currently Researcher at the Centre for Image Analysis, Uppsala University, Sweden; Senior Research Associate at Mathematical Institute of the Serbian Academy of Sciences and Arts, Serbia; and Head of Research at Topgolf Sweden AB, Stockholm, Sweden. His research interests include development of general and robust methods for image processing and analysis.

Nataša Sladoje received the B.Sc. and M.Sc. degrees in Mathematics from the Faculty of Science, University of Novi Sad, Serbia, in 1992 and 1998, respectively, and the Ph.D. degree in Computerized Image Analysis from the Centre for Image Analysis, Swedish University of Agricultural Sciences, Uppsala, Sweden, in 2005. She is a Senior Lecturer at the Centre for Image Analysis, Uppsala University, Sweden. Her research interests include theoretical development of image analysis methods with applications in biomedicine and medicine.