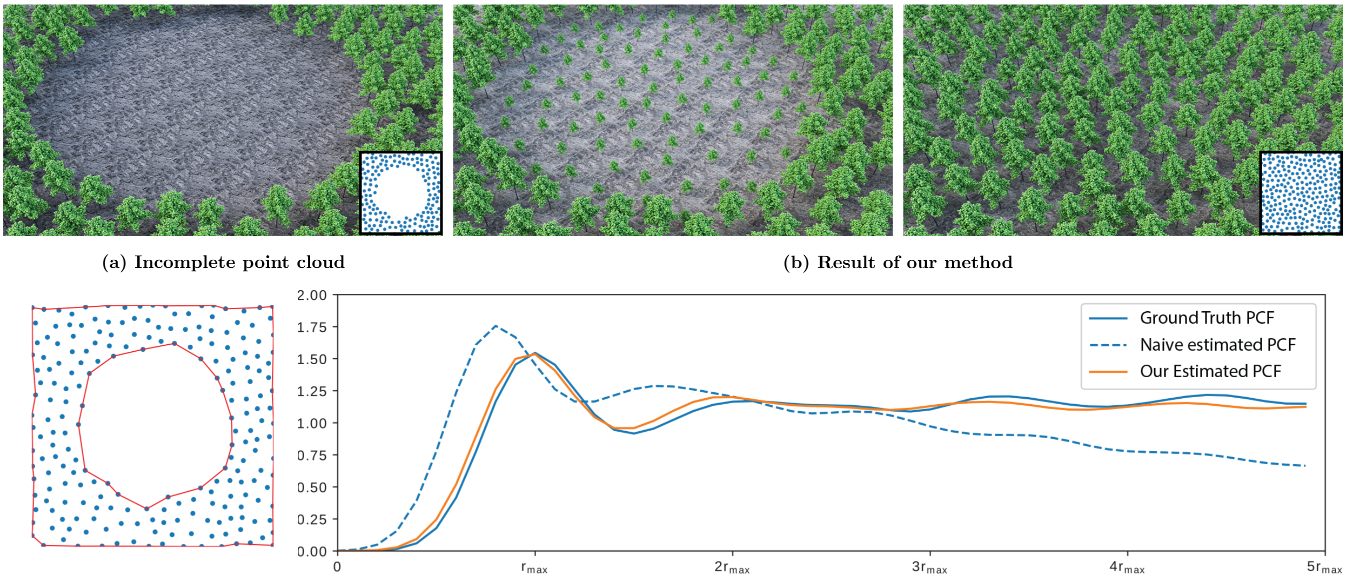
Pair Correlation Functions with Free-Form Boundaries for Distribution Inpainting and Decomposition

B. Nicolet[1], P. Ecormier-Nocca[1], P. Memari[1] M.-P. Cani[1]

[1]École Polytechnique, CNRS (LIX), IP Paris



(a) Incomplete point cloud (b) Result of our method

(c) Estimated Boundary (d) Computed PCFs

Figure 1: *Application of extended PCFs to the inpainting of distributions, illustrated by the re-planting of a partially destroyed forest (a,b). From an incomplete dataset (a), our method accurately estimates free-form boundaries (c) and compensates from missing points when learning PCFs (d). This allows us to restore the perceived initial distribution at the synthesis stage (b).*

Abstract

Pair Correlation Functions (PCF) have been recently spreading as a reliable representation for distributions, enabling the efficient synthesis of point-sets, vector textures and object placement from examples. In this work we introduce a triangulation-based local filtering method to extend PCF-based analysis to exemplars with free-form boundaries. This makes PCF applicable to new problems such as the inpainting of missing parts in an input distribution, or the decomposition of complex, non-homogeneous distributions into a set of coherent classes, in which each category of points can be studied together with their intra and inter-class correlations.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Boundary representations

1. Introduction

The analysis of distributions is a recurrent focus in computer graphic, motivated by a variety of applications, from texture synthesis to sample generation in Monte Carlo rendering, through objects placement from exemplars. Recent methods have relied on Pair Correlation Functions (PCF), which express the density of a

DOI: 10.2312/egs.20201025

point's neighborhood as a continuous function of distance, in order to increase robustness compared to histograms and enable the use of gradient descent at the synthesis stage.

PCF computations require careful handling of the boundary of the analyzed point-set, since point neighborhoods are biased near boundaries. The standard solution is to restrict the analysis to a regular sub-domain, such as a rectangle or a square, enabling to quantify and compensate for the decrease of neighbors. Point cloud data, however, often comes in arbitrary shapes and one cannot always afford discarding some of the points from a small exemplar.

In this paper, we extend PCF analysis to exemplars with free-form boundaries. We introduce a triangulation-based local filtering mechanism to identify boundaries within the input, enabling us to compensate the associated bias. This framework allows us to apply PCF analysis to new problems, such as using an in-painting method to restore distributions with missing data, or decomposing an heterogeneous distribution into a given number of relevant subsets, for which intra-class and inter-class correlations can then be analyzed. We achieved this by applying a standard clustering algorithms in PCF space, while relying on our extended PCFs to yield effective results regardless of the underlying domain.

Our main contributions are:

- An efficient extension of PCF analysis to free form boundaries.
- Methods for reconstructing data regardless of arbitrary domains or missing information, and for decomposing an unlabeled point-sets into relevant sub-distributions.

2. Related Work

Signal decomposition using classical approaches such as ICP, and signal in-painting using wavelets, are commonly used nowadays in several application frameworks from image or audio processing to astrophysics [NSN*19]. However, complex geometric patterns and distribution decomposition have received little attention so far. A commonly considered problem is the separation of homogeneous clusters, where the heterogeneity of a point data set is classically determined based on scale-related functions [BES73,Rip77] or summary indices [PPS06]. A limited case of decomposition of repulsive clusters was recently addressed [SP19]. The closest strategy to ours may be [PGMZ12] which introduces a multi-scale decomposition of point-sets based on the k-th nearest distance transformation. A reversible jump Markov Chain Monte Carlo strategy and a stepwise Expectation-Maximization algorithm are used to progressively divide the data into distinct components. Although comparable to a dual wavelet transform in signal processing, this method has a high complexity and is heavy to use.

Our work rather builds on Pair Correlation Functions (PCFs), developed in computer graphics for the analysis and synthesis of distributions [ÖG12, RÖG17, ENMGC19]. These methods, as well as point-processes based on discrete histograms [EVC*15], are restricted to exemplars conveniently located in a simple shape such as a square, enabling them to compensate the bias due to borders. This has limited their use so far for real-data analysis, where the input may have an arbitrary shape and include holes, and made their application to the decomposition problem difficult.

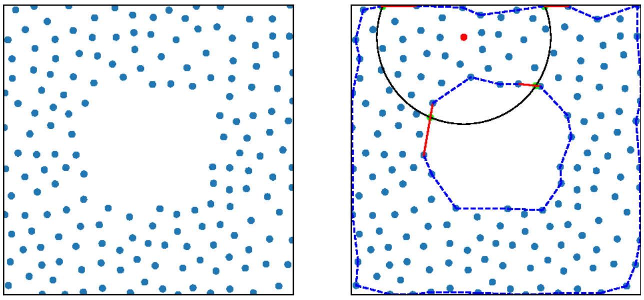


Figure 2: *Incomplete point-set (left) and the extracted free-form boundary (dotted lines, right). For computing the PCF at the red point, we sum the angles for which each circle lies inside the domain (i.e. between green points on red edges).*

3. Free-form PCFs

In this section, we introduce a new simple and intuitive solution for PCF analysis with free boundaries. We rely on the PCF formulation from [ÖG12], as follows. For a given point P_i in a set $P_0, \ldots, P_n$, the PCF is defined for each radius r as:

$$PCF(P_i, r) = \frac{w_i(r)}{A_r n} \sum_{j \neq i} k_\sigma(r - d_{ij}), \qquad (1)$$

where A_r is the area of the ring of radius r and width 1, and d_{ij} is the distance between points P_i and P_j. The Gaussian Kernel $k_\sigma(x) = \frac{1}{\sqrt{\pi}\sigma} e^{-x^2/\sigma^2}$ is used to ensure robustness, centered at distance r around the reference point. The PCF of a point-set is defined as the mean of the individual PCFs of all points. In Eq. 1, $w_i(r)$ is a normalization term used to compensate the influence of the boundary. The other notations are the same as described in [ÖG12]. Traditionally in [EVC*15], this term is computed as:

$$w_i(r) = \frac{2\pi r}{l_i(r)} \qquad (2)$$

where $l_i(r)$ is the length of the circle centered in P_i of radius r contained inside the domain. This weight is designed for square domains, and accounts for missing points on the portions of circles outside them. Our goal is to extend this to free-form boundaries.

3.1. Compensation of missing points

To generalize the weighing term in Eq 2, we first approximate the boundary of the input by computing its Delaunay triangulation and filtering the longest edges. We do so by going through every vertex and discarding the incident edges considered too long according to a local criterion: only the edges shorter than t times the shortest edge are kept, where t is a user-defined threshold. In our implementation, we found that $t = 3$ works well, even for non homogeneous point clouds (e.g. white noise). We then extract the boundary of the point set from the resulting triangulation, by stamping the edges contained in each triangle and filtering out those stamped more than once. Finally, we compute the intersection points between the circle of radius r and the estimated boundary, and the corresponding angles $\Delta\alpha_i$ (see Figure 2). We determine which angular portions lie inside the domain by selecting a point on the current circle with

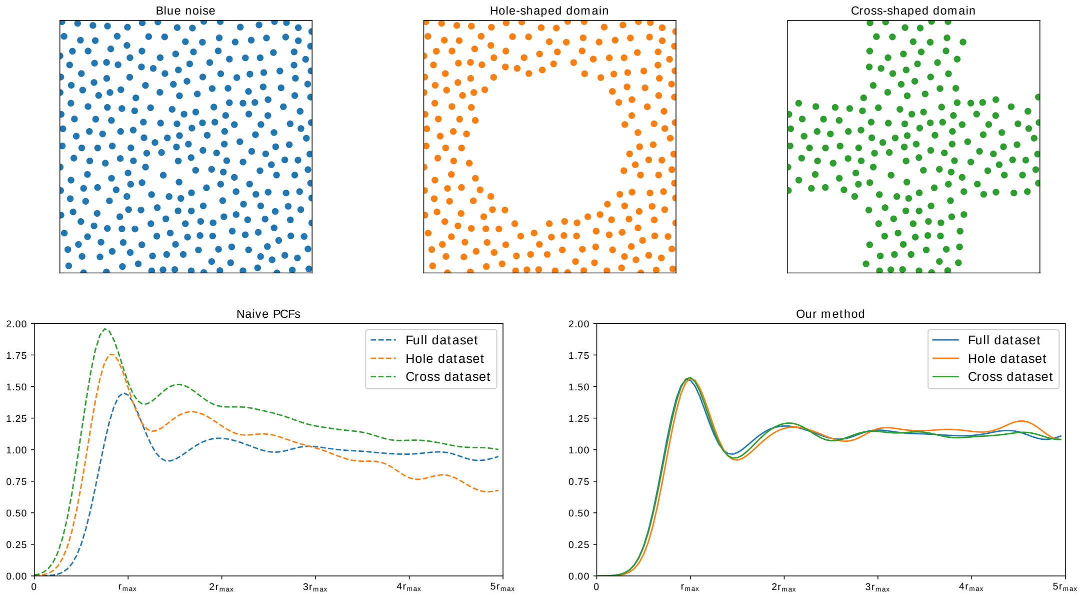


Figure 3: *Analysis of complete vs. incomplete data-sets. Contrary to standard PCFs (bottom left), we get consistent curves (bottom right). This is the desired result since the distribution did not change - the domain only was modified.*

an angle only slightly larger than one of the computed angles, and determine whether it lies inside the boundary. The new weighing term is set to:

$$w_i(r) = \frac{2\pi}{\sum_i \Delta\alpha_i} \tag{3}$$

4. Results and Applications

4.1. Quantitative results

Figure 3 compares our new PCFs with standard ones, on complete vs incomplete exemplars, showing that the mean PCF we extract does not depend on the boundary anymore. In addition, we computed the average standard deviation of individual point PCFs for these three data-sets, shown in Figure 4. The lower standard deviation for our method further confirms its relevance to handle arbitrary domains: While results are similar for the first exemplar (a), we significantly reduce variance in complex cases (25% decrease for cross-shape (c)).

In terms of efficiency, our method is indeed slower than regular PCFs due to the need to compute a Delaunay triangulation, to filter triangles out, and to compute the weight of each point. In practice, we achieve around 9 seconds for 300 points in a unit square domain (a), 7 seconds for the domain with a hole (b), and 6 seconds for the cross-shaped domain (c), compared to between half and a quarter of a second for standard PCFs. Our algorithm was implemented on the CPU in Python and could be optimized. Timings were computed on an Intel® Core i5 clocked at 3.3GHz with 8GB of RAM.

4.2. Application to Distribution In-painting

Filling holes in a distribution with missing data is an immediate application, and a good illustration of our method. After computing

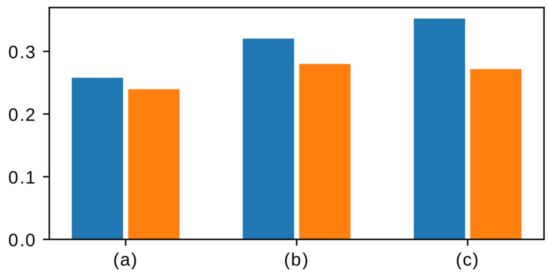


Figure 4: *Comparison of the average standard deviation of individual PCFs for regular (blue) vs. our PCFs (orange) on full (a), hole-shaped (b) and cross-shaped (c) domains.*

boundary-independent PCFs, we simply add samples to the input point-set such that each added point approximately maintains the extracted PCF, and then fine-tune their position using gradient descent (similarly to the synthesis stage in [ENMGC19]). Figure 1 shows that a dataset with significant missing data is sufficient to successfully recover an unbiased representation of the input, and then seamlessly reconstruct missing parts.

4.3. Application to Distribution Decomposition

Our method can also be used to decompose a complex, unlabeled point-set into coherent classes in terms of distribution, also enabling to analyze extra-class correlations. We use our boundary-handling method to compute the PCFs of individual points, and then cluster them by applying k-means in PCF space, where the number k of clusters is preset by the user. Two clustering results are

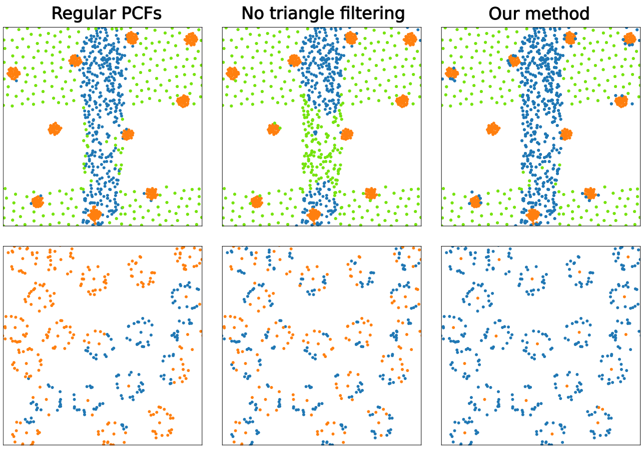


Figure 5: *Clustering results on 2 datasets, using regular PCFs, our method without, and with local triangle filtering. The colors represent the classes assigned to each sample.*

shown on Figure 5. The use of our PCFs with free-form boundaries allows us to compute a relevant distribution whatever the point position, resulting in robust and consistent clustering results. We show results without the local triangle filtering step (Figure 5, center), to underline its importance in our framework.

5. Discussion & Future Work

While results demonstrated the relevance of our method, the main drawback is the increased computational time. The latter depends in part on the actual length of the boundary, which cannot be avoided. A GPU implementation would be a good extension.

Furthermore, our application to clustering is limited to cases where the user can give a relevant number of target clusters, while the use of distance between PCF curves as a perceptual metric to decompose distributions remains to be validated.

Another possible bias stems from the fact that the PCFs used for clustering were initially computed on the whole dataset, before partitioning. Thus, they can be different from the final PCFs obtained by considering only points within each class. This issue, along with the current necessity to manually choose the number of clusters to extract, could be solved by implementing an iterative clustering method that would incrementally split and merge clusters according to a fitness metric, while making sure to update the computed PCFs as the content of each class changes.

As a more general follow-up, we hope that an improved distribution decomposition method could outperform existing clustering-based approaches for analyzing high-dimensional point-sets. For instance, it could be used in a dimension reduction strategy for multi-label high dimensional point-clouds, where the optimal projection is the one for which the distribution decomposition is compatible with the high dimensional labels (clusters). This should outperform clustering-based approaches aiming at finding non-overlapping aggregated clusters in the projection.

Lastly, our method might also be applicable to image denoising. Indeed, noise typically follows specific distribution laws which differ from the information of the underlying image, and might thus be automatically separated from the input using distribution decomposition tools.

6. Conclusion

We proposed a new method for analyzing point distributions on arbitrary-shaped domains. Our framework improves the computation of a common statistical tool, called Pair Correlation Function (PCF), through an efficient boundary handling. The method relies on a triangulation-based local filtering process to estimate the border of the point set, enabling to weigh the computed functions accordingly. We have successfully applied this method firstly to distributions in-painting, a useful tool in case of missing data, and secondly to the problem of decomposing complex non-labeled point-sets into a given number of relevant distributions.

In the future, we plan to investigate the extension of the method to distributions of higher dimensions. In addition, similar boundary handling can be considered for related approaches such as pairwise differential histograms, as introduced in [WW11] for anisotropic sampling, considering both spatial and angular distributions. Lastly, applying our decomposition method to the analysis of real data, such as retrieving the relevant plants species in an ecosystem from a photograph and studying their correlations, would be an interesting application for this work.

References

[BES73] BESAGE J.: On the detection of spatial pattern in plant communities. *Bull. int. statist. Inst. 45* (1973), 153–158. 2

[ENMGC19] ECORMIER-NOCCA P., MEMARI P., GAIN J., CANI M.-P.: Accurate synthesis of multi-class disk distributions. In *Computer Graphics Forum* (2019), vol. 38, Wiley Online Library. 2, 3

[EVC*15] EMILIEN A., VIMONT U., CANI M.-P., POULIN P., BENES B.: Worldbrush: Interactive example-based synthesis of procedural virtual worlds. *ACM Transactions on Graphics (TOG) 34*, 4 (2015). 2

[NSN*19] NAMMOUR F., SCHMITZ M. A., NGOLÈ F., STARCK J.-L., GIRARD J. N.: Radio astronomical image restoration with shape constraint. In *Wavelets and Sparsity XVIII* (2019), vol. 11138, International Society for Optics and Photonics, p. 111380A. 2

[ÖG12] ÖZTIRELI A. C., GROSS M.: Analysis and synthesis of point distributions based on pair correlation. *ACM Transactions on Graphics (TOG) 31*, 6 (2012), 170. 2

[PGMZ12] PEI T., GAO J., MA T., ZHOU C.: Multi-scale decomposition of point process data. *GeoInformatica 16*, 4 (2012), 625–652. 2

[PPS06] PASCUAL D., PLA F., SÁNCHEZ J. S.: Non parametric local density-based clustering for multimodal overlapping distributions. In *International Conference on Intelligent Data Engineering and Automated Learning* (2006), Springer, pp. 671–678. 2

[Rip77] RIPLEY B. D.: Modelling spatial patterns. *Journal of the Royal Statistical Society: Series B (Methodological) 39*, 2 (1977), 172–192. 2

[RÖG17] ROVERI R., ÖZTIRELI A. C., GROSS M.: General point sampling with adaptive density and correlations. In *Computer Graphics Forum* (2017), vol. 36, Wiley Online Library, pp. 107–117. 2

[SP19] SONG C., PEI T.: Decomposition of repulsive clusters in complex point processes with heterogeneous components. *ISPRS International Journal of Geo-Information 8*, 8 (2019), 326. 2

[WW11] WEI L.-Y., WANG R.: Differential domain analysis for non-uniform sampling. *ACM Transactions on Graphics (TOG) 30*, 4 (2011), 1–10. 4