Higher level spatial analysis of dead pixels on detectors based on local grid geometry

Julia A Brettschneider, Jason W Warnett, Tom E Nichols, Wilfrid S Kendall

Abstract

Let a collection B_i $(i \in I)$ of binary values indexed by a grid $I \subseteq \mathbb{Z}^2$ be a model to describe dysfunctional pixels in images obtained by a detector based technology. While there are obvious global measures to quantify the amount of damage, natural questions also arise around the spatial distribution of the dysfunctional pixels and how observed patterns of dysfunctional pixels may be interpreted.

After modelling occurrences of dysfunctional pixels as a planar point process we develop a higher level approach for analysing their spatial distributions. Key idea is to move from the notion of a dysfunctional pixel to the concept of a *damage event* defined by configurations of dysfunctional pixels using a typology based on local grid geometry. High density regions can be detected using density estimation of the damage event process, so remaining areas becomes suitable candidates for complete spatial randomness. This approach decouples observed damage from the detector resolution prescribed by I and from the exact shape of dysfunctional pixel configurations.

We proposes a detector quality toolkit that allows users to monitor their technology following these principles. The methods allow users of detector based imaging technologies to detect, distinguish and monitor different types of quality damage and to identify the ones linked to specific causes. We apply our methods to a collection of *bad pixel maps* obtained as part of regular monitoring routines of a detector used in X-ray computed tomography.

1 Introduction

Dead pixels have been known to users of many types of detectors for a long time including computer screens, digital photographic cameras, digital video cameras and X-ray detectors. Digital flat detector panels are used, for example, in computed tomography (see e.g. [4] and [3]). Dead pixels are a nuisance for all of these technologies. Damage in X-ray detectors is particularly challenging, because they are very costly: repairs can cost on the order of hundreds of thousands of pounds.

It is therefore of practical interest to establish methods to understand the severity of the damage and determine whether or not, and how, the detector could be used despite some damage. Furthermore, in light of preventing further damage in both a currently used detector and future detectors, there is a central interest in understing the the sources of damage.

In the words of statistical quality control pioneer Walter Shewhart, the objective is to distinguish between special reasons for poor quality and common reasons for poor quality. Roughly speaking, the latter have to be accepted as long as they are

Paper No. 17-02, www.warwick.ac.uk/go/crism

within limits. The former, however, need to be identified and tackled by improving the technology or other parts of the industrial process.

We propose a quality assessment algorithm based on moving beyond the perspective of an individual damaged pixel to a perspective of collections of damaged pixels. This change of perspective helps identifying the actual sources of damage rather than just observing their effect. The property of complete spatial randomness can be interpreted as what Shewhart would call common causes of poor quality. Hence our focus is on determining whether or not, or where, this property holds.

This change of perspective is done in two ways. Firstly, we move from the pixel process, that is, a point process describing individual dysfunctional pixels, to an event process, that is, a point process consisting of damage events defined by collections of connected dysfunctional pixels. Simple definitions based on grid geometry are used in the construction. On the higher level provided by the event process it becomes more meaningful to associate complete spatial randomness of the corresponding point process with damage occurring at random, whereas deviations from this property indicate that damage was caused by localised specific technical problems.

Secondly, we use density fitting to the event process to determine areas of elevated damage. The latter can be removed, this enabling us to study the remaining parts of the detector independently of such local damage.

As an illustration of our methods we analyse a collection of bad pixel maps taken on the same digital X-ray detector over a period of seven months (see Section 4.1). An initial round of exploratory data analysis in [2] sheds light on the different types of dysfunctionality that can be encountered for pixels and it guides our choice of spatial models in Section 2. Practical applications are discussed in Section 4.3.

2 Spatial models

2.1 Pixel process and clusters

Dysfunctional pixel location are described by a collection B_i $(i \in I)$ of binary values indexed by a finite grid $I \subset \mathbb{R}^2$. A common example for I is a grid rectangle $[1, \ldots, n_1] \times [1, \ldots, n_2]$.

The collection B_i $(i \in I)$ can also be understood as a realisation of a point process X. This X can be represented either as a random binary function $X : I \to \{0, 1\}$ or as a subset $\{i \in I \,|\, X_i = 1\} \subseteq I$. Note that the underlying point process X only places points on the discrete grid $I \subset \mathbb{Z}^2$. However, it could be an approximation to a point process defined on a subset of $\mathbb{R}^2$.

For $z \in \mathbb{Z}$ Let π_z define the *projections* on z by

$$\pi_1^z(i) = (z, i_2) \text{ and } \pi_2^z(i) = (i_1, z) \qquad (i \in I)$$

They allow us to represent the projection of a set $J \subseteq I$ on the vertical line $\{(x, y) \mid y \in \mathbb{Z}\}$ through x by $\pi_1^x(J)$. Similarly, the projection on the horizontal line $\{(x, y) \mid x \in \mathbb{Z}\}$ through y is given by $\pi_2^y(J)$.

For any $J \subseteq I$ we can measure the expansion in each of the dimensions by

$$\text{width}(J) = \max\big\{|j_1 - j_1^{'}| \,\big|\, j, j^{'} \in J\big\} \text{ and } \text{height}(J) = \max\big\{|j_2 - j_2^{'}| \,\big|\, j, j^{'} \in J\big\}.$$

We see that for any $y \in \mathbb{Z}$, $\text{width}(J) = \big|\pi_2^y(J)\big|$ and for any $x \in \mathbb{Z}$, $\text{height}(J) = \big|\pi_1^x(J)\big|$.

We further define statistics for the separate dimensions $d \in \{1, 2\}$

$$\begin{aligned} \text{mode}_d(J) &= \text{mode}\big(\{j_d \mid j \in J\}\big) \\ \text{mean}_d(J) &= \text{mean}\big(\{j_d \mid j \in J\}\big) \\ \text{median}_d(J) &= \text{median}\big(\{j_d \mid j \in J\}\big) \end{aligned}$$

For $i = (i_1, i_2) \in I$ we define its nearest neighbour set as

$$\mathcal{N}(i) = \big\{j \in I \,\big|\, |i_1 - j_1| + |i_2 - j_2| = 1\big\}$$

("rook" neighbourhoods). For simplicity, in this paper, we always use this definition, but it could be replaced by an alternative definition of neighbourhood. Another common choice would be the set $\{j \in I \mid |i_1 - j_1| = 1 \vee |i_2 - j_2| = 1\}$ ("queen" neighbourhoods). that also includes the diagonally adjacent pixels of i. An n-tupel $(i^{(1)}, \ldots, i^{(t)})$ of elements of I is called an $\mathcal{N}$*-path* if $i^{(s+1)} \in \mathcal{N}(i^{(s)})$ for all $s \in \{1, \ldots, t-1\}$. Two points i and j are called $\mathcal{N}$*-connected* if there is an $\mathcal{N}$-path between them. A set $\mathcal{C} \subseteq I$ is called an$\mathcal{N}$*-cluster* if all $i, j \in \mathcal{C}$ are $\mathcal{N}$-connected. For the sake of brevity, $\mathcal{N}$ may be dropped from the notation.

We now introduce a few categories to distinguish different types of clusters. A trivial case for a cluster is $|\mathcal{C}| = 1$, in which case it is called a *singleton.* A cluster $\mathcal{C} \subseteq I$ is called a *double* if $|\mathcal{C}| = 2$ and a *triplet* if $|\mathcal{C}| = 3$.

If $|\mathcal{C}| \geq 4$ we distinguish between *lines* and *large clusters.* A narrow definition of a line would be a one-pixel wide straight $\mathcal{N}$-path of some minimal lengths $\alpha_1 \in \mathbb{N}$. In light of practical applications we allow small deviations from this condition by tolerating a few pixels adjacent to the actual line and some extra ones at the end. This is formalised by using parameters $\alpha_2 \geq 1$ and $\alpha_3 \in \mathbb{N}_0$ in the following conditions involving the parameter $\alpha = (\alpha_1, \alpha_2, \alpha_3)$. A cluster $\mathcal{C} \subseteq I$ is called a *vertical* α*-line* if

$$|\mathcal{C}| \geq \alpha_1, \text{ width}(\mathcal{C}) \leq 3 \text{ and } |\mathcal{C}| \leq \alpha_2 \,\big|\big\{i \in \mathcal{C} \text{ with } i_1 = \text{mode}_1(\mathcal{C})\big\}\big| + \alpha_3,$$

while it is called a *horizontal* α*-line* if

$$|\mathcal{C}| \geq \alpha_1, \text{ height}(\mathcal{C}) \leq 3 \text{ and } |\mathcal{C}| \leq \alpha_2 \,\big|\big\{i \in \mathcal{C} \text{ with } i_2 = \text{mode}_2(\mathcal{C})\big\}\big| + \alpha_3.$$

$\mathcal{C}$ is called an α*-line* if it is either a horizontal or a vertical α-line. (For example, $\alpha_1 = 4$, $\alpha_2 = 1.1$, and $\alpha_3 = 2$ means lines have to be at least 4 pixels long, and up to

10% additional pixels along the sides and two more at the ends are acceptable.) An α suitable for the context can be chosen when the methods are applied to data and should remain fixed through the subsequent analysis. We then simply refer to an α-line as a *line*.

If $|\mathcal{C}| \geq 4$ and it is not a line then it is called a *large cluster*.

2.2 Event process

A weakness of the pixel process introduced in the last section is its dependency on the exact placements of damages responsible for rendering pixels dysfunctional. A change of location of the damaged area can lead to a different set of affected pixels. If the area is not rationally invariant, then a rotation can have that effect as well. Examples for these scenarios are shown in Figure 1.

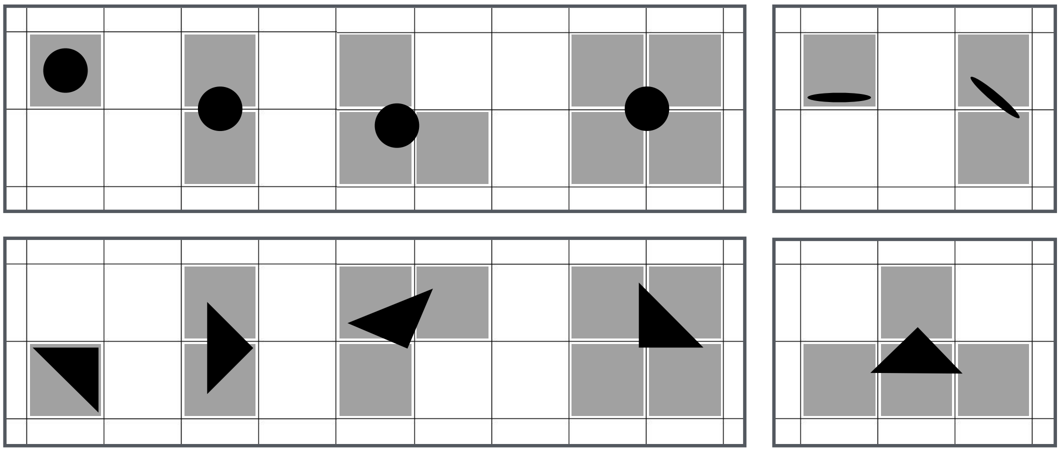


Figure 1: **Damage covering multiple pixels depending on their shape and placement relative to the grid.** Examples for events symbolising damages are shown in solid black shapes and resulting dysfunctional pixels are shown in solid grey squares. The top left series involving a circular damage event demonstrates how its exact location relative to the grid lines, can cause 1, 2, 3 or 4 dysfunctional pixels. The top right shows how for a needle shaped damage event a change of angle can make the difference of causing 1 or 2 dysfunctional pixels. The bottom left series involving a triangular damage event shows that a rotation can make the difference between 1, 2, 3 or 4 affected pixels. The bottom right shows a change of location can result in a different arrangement of the 4 dysfunctional pixels.

To overcome these difficulties, we convert the pixel process X into a higher level process Y, which we call the *event process.* Y is a marked point process with points determined by clusters of pixels from the original point process X. Each cluster $\mathcal{C}$ of X is represented by one point $i^r(\mathcal{C}) \in I$ and is marked based on shape classification: If $\mathcal{C}$ is an α-line, $i^r(\mathcal{C})$ is defined as one of the endpoints (which one can be chosen according to the specific application). For other clusters, i^r is chosen by some defined procedure so as to represent a centre of the cluster. We will choose for $i^r(\mathcal{C})$ the point

$$i^{\text{median}}(\mathcal{C}) = \big(\rho(\text{median}_1(\mathcal{C})), \rho(\text{median}_2(\mathcal{C}))\big). \tag{1}$$

where ρ is the rounding function. Alternative definitions of i^r could be used, depending on the application. For example,

$$i^{\text{mean}}(\mathcal{C}) = \big(\rho(\text{mean}_1(\mathcal{C})), \rho(\text{mean}_2(\mathcal{C}))\big). \tag{2}$$

Also, ρ could be replaced by another appropriate integer valued function on $\mathbb{R}$. However, while replacing ρ may result in a different choice for i^r, the difference will typically be small enough not to matter in practice.

For singletons, i^{median} and i^{mean} are always the pixel itself and for doubles they are one of two pixels. For triplets arranged in a straight line, i^{median} and i^{mean} are the middle pixel. In general, i^r will capture the concept of being a centre of the cluster, but details depend on the choice of the definition and the cluster shape. In particularly, due to rounding and non-convexity, is is possible for i^r not to be part of the cluster. Since it is more important for i^r to robustly capture the location where the bulk of the pixels of a cluster is located than doing so with precision, the use of the median (1) is usually preferable to the mean (2). Specifically, the use of the median has the advantage of limiting the effect of an individual long "hair" sticking out of a cluster as shown in Figure 2 on the choice of the point i^r.

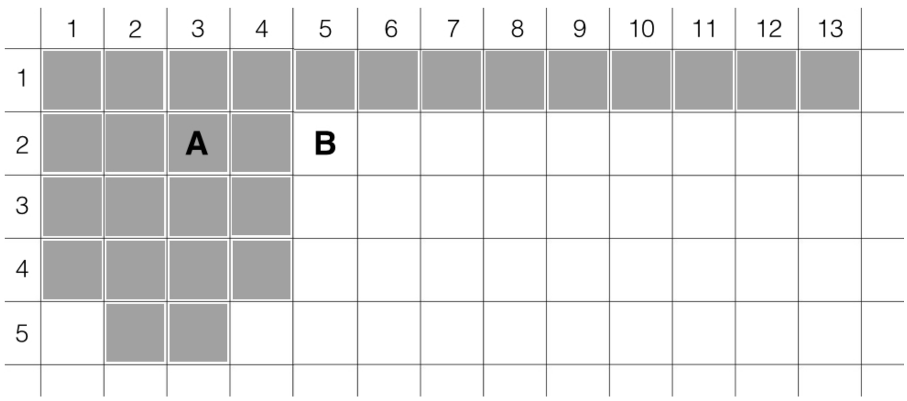
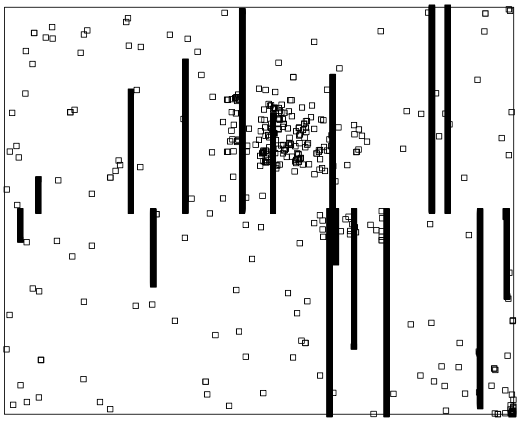
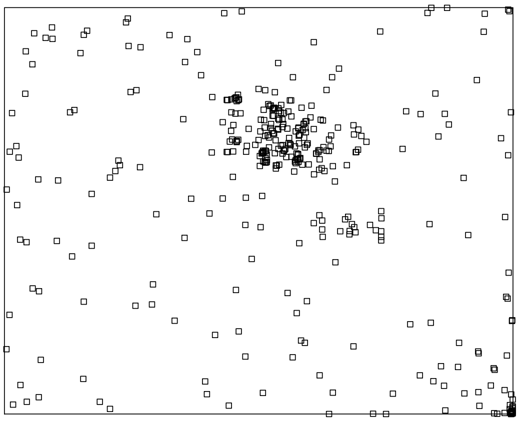


Figure 2: **Example for a pixel cluster with a "hair" sticking out.** The first row of pixels in the cluster $\mathcal{C}$ shown extends well beyond the other rows pulling $\text{mean}_1(\mathcal{C})$ to the right resulting in $i^{\text{mean}}(\mathcal{C}) = (5, 2)$ marked as Point B ($\rho(\text{mean}_1(\mathcal{C}) = \rho(4.\overline{6}) = 5$, $\rho(\text{mean}_2(\mathcal{C})) = \rho(2.185185) = 2$). In contrast, $i^{\text{median}}(\mathcal{C}) = (3, 2)$ marked as Point A.



(a) Pixel process



(b) Event process (marks not visualised)

Figure 3: **Pixel process and event process.** Applying the appropriate rules, configurations of connected pixels in the pixel process are reduced to one point per configuration when constructing the event process. In the example, lines provide the most striking instances of damage, but there is also damage in corners and in some other areas.

3 Quality assessment tools

There is a variety of objectives in the quality assessment of detectors which can be associated with suitable statistical measures. Our approach has several components based on global information, local configurations and spatial distributions of these. Both pixel level and event level information are used for a variety of scores we propose for usage in the context of quality assessment.

We uses very simple scores for rating overall detector quality:

$$\begin{aligned} \textit{functional pixel percentage} &= \#\text{functional pixels}/|I| \\ \textit{damage events count} &= \big|\{\mathcal{C} \,|\, \mathcal{C} \text{ is a cluster}\}\big| \end{aligned}$$

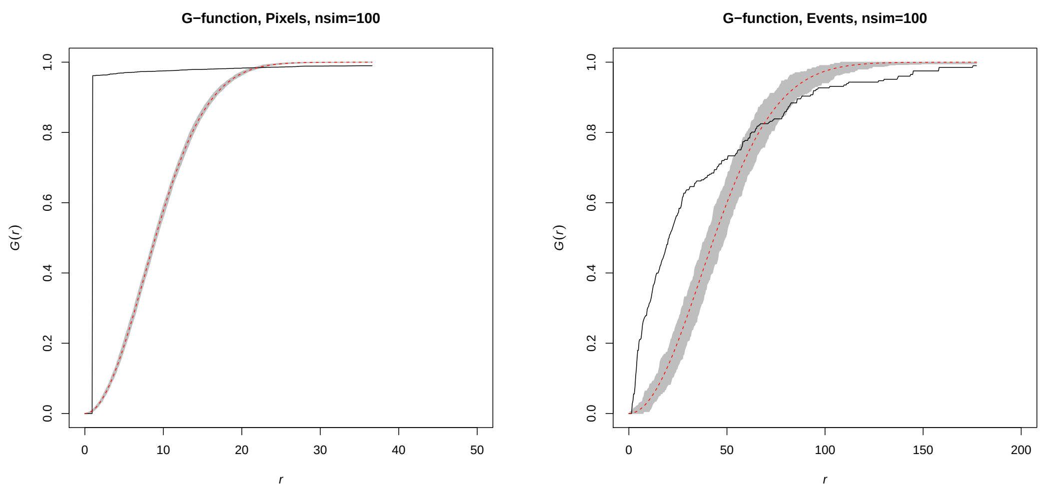
Our local approach involves spatial analysis of the distribution of damage events rather than individual dysfunctional pixels. Based on the classification in Section 2.1, dysfunctional pixels belong to five categories: singletons, doubles, triplets, large clusters and lines and we summarise this using the simple scores listed below.

$$\begin{aligned} &\textit{singleton count} = \big|\{\mathcal{C} \,|\, \mathcal{C} \text{ is singleton}\}\big| \\ &\textit{line count} = \{\mathcal{C} \,|\, \mathcal{C} \text{ is a line}\} \\ &\textit{non-line cluster count} = \big|\{\mathcal{C} \,|\, \mathcal{C} \text{ is singleton, double, triplet or large cluster}\}\big| \\ &\textit{median line length} = \text{median}\{|\mathcal{C}| \,|\, \mathcal{C} \text{ is a line}\} \\ &\textit{median cluster size} = \text{median}\{|\mathcal{C}| \,|\, \mathcal{C} \text{ is singleton, double, triplet or large cluster}\} \end{aligned}$$

Apart from counting damage events and measuring their average size, we need to summarize spatial distribution. Let Z be a point process with state space I and (globally measured) intensity λ. We distinguish between *locations,* that is any element in I, and *points*, that is a location contained the realisation of the process in question. A central question is whether Z process has the property of *complete spatial randomness* (short: *CSR*), which means that the points are distributed independently and homogeneously over the state space I, such as for the homogeneous Poisson process.

The *nearest neighbour function* G is the cumulative distribution function of the distance from an arbitrary point to its nearest point. Under CSR, $G(r) = 1 - \exp(-\lambda\pi r^2)$. The *empty space function* F is the cumulative distribution function of the distance from an arbitrary location to its nearest point. Under CSR, $F(r) = 1 - \exp(-\lambda\pi r^2)$. (The two measures typically differ if CS does not hold.)

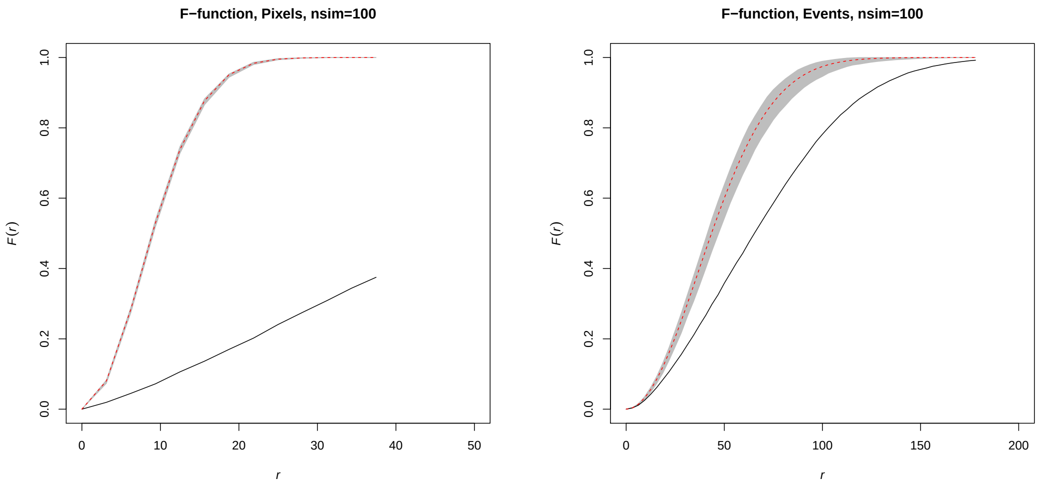
Ripley's K-function calculates the expected number of points as a function of the distance r for any point, that is, $K(r) = \lambda^{-1}E[N_0(r)]$, where $N_0(r)$ is the number of points up to a distance of r from an arbitrary point of the process. It provides a measure for the interaction between the points of the process and helps identifying and competition at different scales. Under CSR, $K(r) = \pi r^2$.



(a) G-function of the pixel process

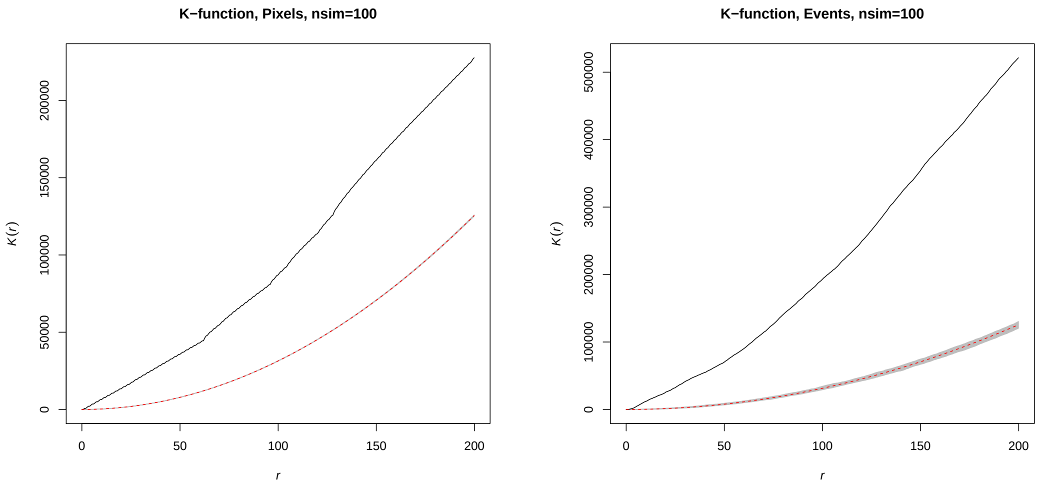
(b) G-function of the event process

Figure 4: **G-function.** Empirical processes (black), under CSR (red) with confidence bands (grey); horizontal scales differ. For the pixel process X the empirical G-function increases very steeply for small distances r indicating the presence of areas with higher abundance of points than the global density would suggest. For the event process Y the empirical G-function is less steep, but still increases much more than its CSR counterpart.



(a) F-function of the pixel process (b) F-function of the event process

Figure 5: **F-function.** Empirical processes (black), under CSR (red) with confidence bands (grey); horizontal scales differ. For the pixel process X the empirical F-function increases much slower than its CSR counterpart indicating the presence of areas with lower abundance of points than the global density would suggest. For the event process Y this discrepancy is smaller but still noticable.



(a) K-function of the pixel process (b) K-function of the event process

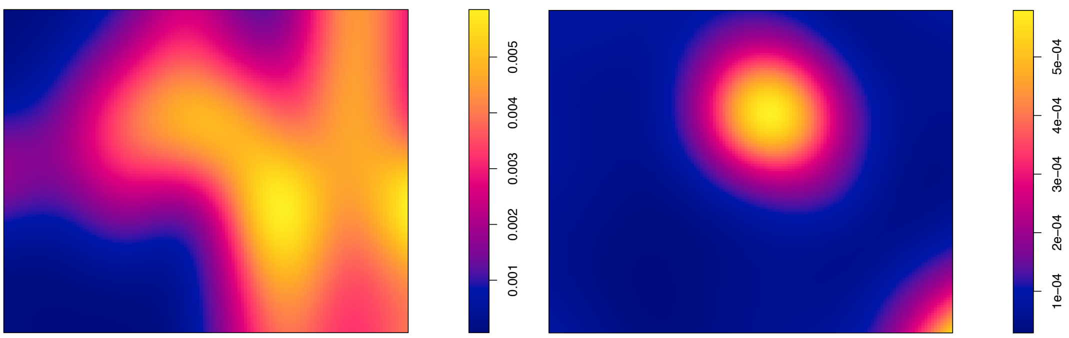
Figure 6: **K-function.** Empirical processes (black), under CSR (red) with confidence bands (grey); vertical scales differ. For the pixel process X the empirical K-function is linear for small distances r suggesting a dominating effect of the lines. For the event process Y the increase starts with a tiny delay, but the discrepancy from K-function under CSR is large. Note the the K-function is normalised with λ, which is much higher for the pixel process than for the event process due to lines and large clusters containing a lot of pixels. This explains why the empirical K-function for the pixel process is smaller in absolute terms than the one for the event process.

The G-, F- and K-function are used in spatial statistics to investigate characteristics of point processes at different distance scales. More details about these functions, their estimators and their R-implementation can be found for example in Chapter 7 of [1] and in [5].

Relevant R packages are `spatstat, sp` and dependencies. Once the data has been imported into a `ppp`-object (defined in `spatstat`) it is straight forward to plot images of the point pattern highlighting the events by out-of-scale plotting characters. Graphs of the G-, F- and K-function typically show the empirical function (black) and the theoretical one (red) under CSR with confidence intervals (grey) calculated by the R-function `envelope()` using simulations. We apply them to both the pixel process and the event process using the same data as in Figure 3.

Figures 4(a), 5(a) and 6(a) illustrate how all three functions indicate that the pixel process X is not CSR for our chosen dataset. Reasons for this surely include the presence of lines and other $\mathcal{N}$-clusters. Figures 4(b), 5(b) and 6(b) show that switching from X to the event process Y can substantially reduce the discrepancy to their behaviour under CSR. However, it also shows that there can be additional quality issues besides $\mathcal{N}$-clusters.

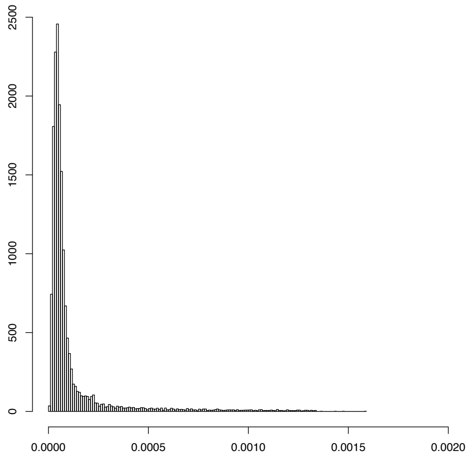
In the example data used to create these figures, the remaining deviations are due to damage in corners and some areas with more damage in a little to the top right of the centre (see Figure 3). These are areas with strong aggregation of points, but they are not $\mathcal{N}$-clusters and therefore they not altered by moving from the pixel process to the event process. We will suggest a method to isolate these areas.



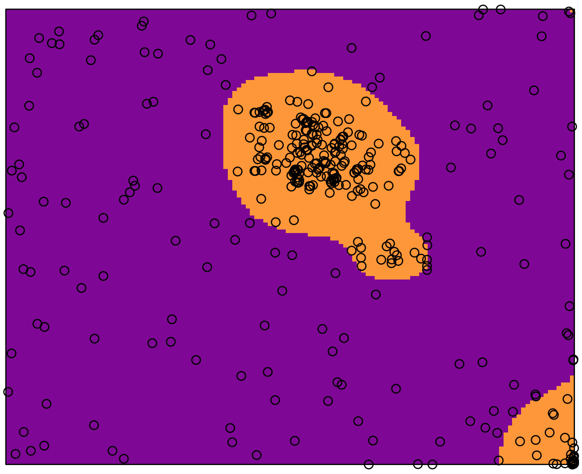
(a) Fitted event pixel density ψ_X (b) Fitted event process density ψ_Y

Figure 7: **Densities for pixel and event process.** This illustration uses the same data as for Figure 3. Fitting a density ψ_X on the level of the pixels process X leads to it being dominated by lines as shown in the left figure. In the event process Y, lines are reduced to points leaving the fitted density ψ_Y unaffected.

We fit a density ψ using a Gaussian kernel. A constant density would mean the damages are broadly evenly spread over the entire index space I. Due to imposed smoothness, individual pixels and sufficiently small clusters will not affect the density.



(a) Histogramm of fitted event process density



(b) Thresholding event process density

Figure 8: **Event process density.** The distribution of the event process density shown in (a) has a long right tail stemming from areas of elevated damage. These can be isolated by applying a threshold as illustrated in (b) using $\delta_0 = 1.5$. In this example these are the area close to the bottom right corner and an area in the top right of the centre.

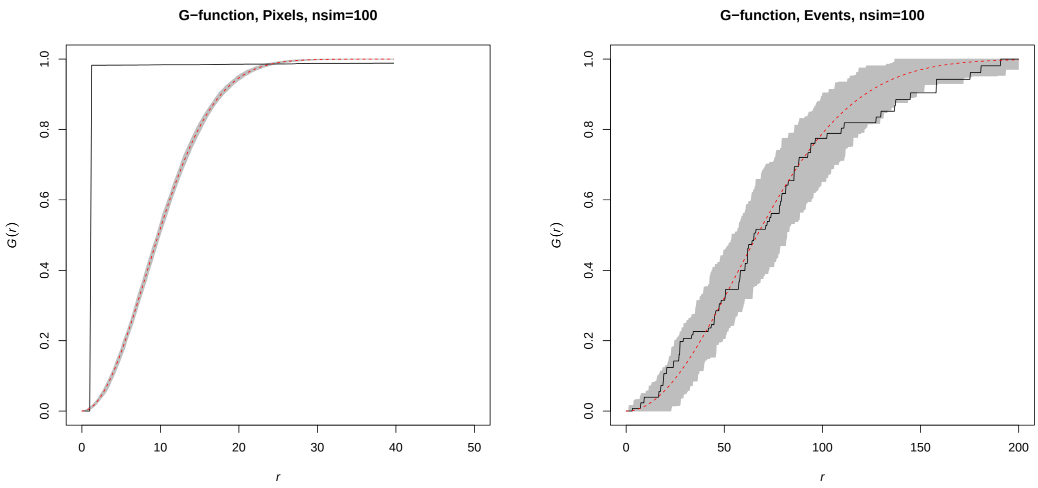
However, lines will dominate the fit of the density ψ_X of the pixel process X whereas the density ψ_Y fitted to the event process Y is unaffected by them as illustrated in Figure 7. For quality assessment purposes we will only use ψ_Y.

We use ψ_Y to identify *areas of elevated damage* thresholding at

$$\delta = q_u(\psi_Y) + \delta_0 \cdot \mathrm{IQR}(\psi_Y), \tag{3}$$

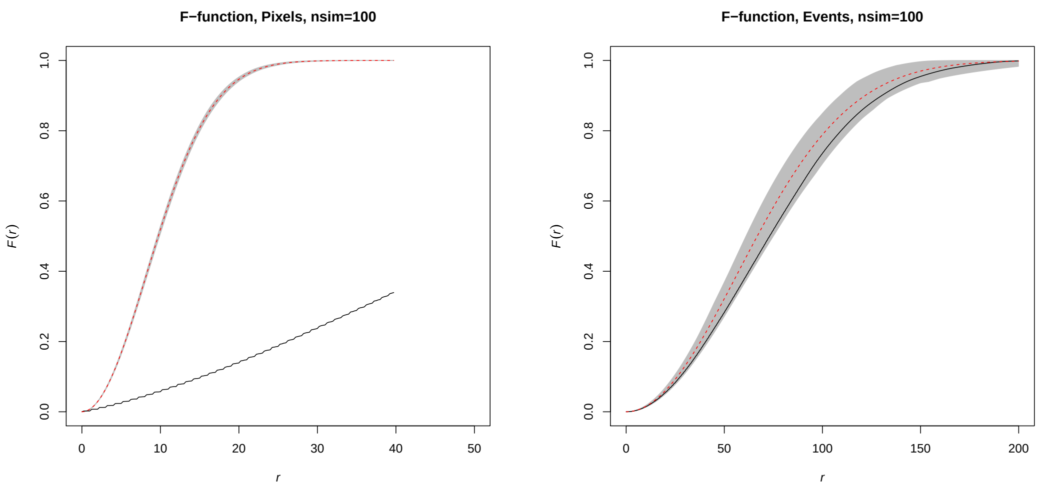
where q_u is the upper quartile, IQR the interquartile range and δ_0 a constant chosen in the context of the data being analysed.

Let D be the union of all areas of elevated damage and let X_{I-D} and Y_{I-D} be the pixel process and the event process restricted to $I - D$. Our main goal is to examine Y_{I-D} for CSR using the G-, F- and K-functions. We also show the corresponding plots for X_{I-D} as that confirms the roles of lines and large clusters for the behaviour of these functions.



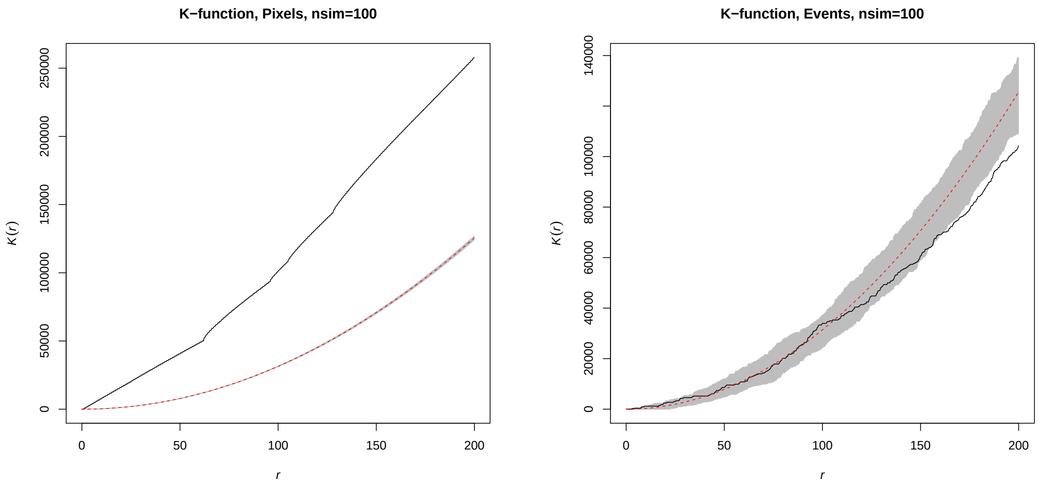
(a) G-function of the restricted pixel process (b) G-function of the restricted event process

Figure 9: **G-function after removing areas of elevated damage.** Empirical processes (black), under CSR (red) with confidence bands (grey) derived under CSR assumptions; horizontal scales differ. For the pixel process X_{I-D} the empirical G-function increases very steeply for small distances r, similarly to Figure 4(a). For the event process Y_{I-D} the empirical G-function is within the confidence bands of its CSR counterpart.



(a) F-function of the restricted pixel process (b) F-function of the restricted event process

Figure 10: **F-function after removing areas of elevated damage.** Empirical processes (black), under CSR (red) with confidence bands (grey) derived under CSR assumptions; horizontal scales differ. For the pixel process X_{I-D} the empirical F-function increases very slowly for small distances r, similarly to Figure 5(a). For the event process Y_{I-D} the empirical F-function is within the confidence bands of its CSR counterpart.



(a) K-function of the restricted pixel process (b) K-function of the restricted event process

Figure 11: **K-function after removing areas of elevated damage.** Empirical processes (black), under CSR (red) with confidence bands (grey) derived from CSR assumptions; vertical scales differ. For the pixel process X_{I-D} the empirical K-function is entirely dominated by the effect of lines creating a linear dependency for small r, similarly to Figure 6(a). For the event process Y_{I-D} the empirical K-function is within the confidence bands of its CSR counterpart for all distances smaller than r around 150. The small deviations for larger distances are likely due to edge issues typical for the K-function.

Figures 9(b), 10(b) and 11(b) show that switching from the pixel process X to the event Y process in combination with removing areas of elevated damage can result in a CSR process.

In practice, this example shows that, after removing some particularly bad areas, damage location on the detector follow CSR.

One could also ask which type of damage effects the behaviour of these functions more: the $\mathcal{N}$-clusters or the areas of elevated damage? More specifically, initially, it is unclear whether the deviations of the pixel process X from CSR observed in Figures 4(a), 5(a) and 6(a) are more due to $\mathcal{N}$-clusters or to areas of elevated damage. A more detailed comparison can settle this question. In this data example, removal only of the areas of elevated damage has a smaller effect (see Figures 9(a), 10(a) and 11(a)) than removal of the $\mathcal{N}$-clusters by switching to the event process Y (see Figures 4(b), 5(b) and 6(b)). This relative balance could be different in a different data example.

Summarising the steps above, we propose the following quality assessment algorithm starting with a binary matrix indicating dysfunctional pixels.

Quality assessment for detectors

1. Construct the pixel process X and calculate the *functional pixel percentage.*
2. Identify clusters and calculate *damage event count, line count, median line length, cluster count* and *median cluster size.*
3. Define an event process Y by restricting clusters and lines to one point each.
4. Fit a density ψ_Y to Y and identify areas with elevated damage by thresholding. If they are close to the edges, the detector can be cropped and is still usable. Otherwise the urgency of the repair will depend on the severity of the damage.
5. Determine whether Y is completely spatially at random outside areas with elevated damage by applying the G-, F- and K-functions to the restricted event process Y_{I-D}.

4 Application to computed tomography data

4.1 Technology

X-ray detectors play a central role in computed tomography; see e.g. [3]. The data used in this case study was collected with the XRD 1621 detector manufactured by *PerkinElmer* for use in X-ray machines. As detailed in the manual [6], it consists of a sensor and its electronics, with the latter placed on the perimeter of the active sensor, out of direct path of the beam. The user needs to block the radiation by lead shielding to avoid damage of the electronics and to adjust the *field of view (FOV).* The flat panel sensor of the detector is fabricated using thin film technology based on amorphous silicon technology which detects visible light. The incident X-rays are converted by the scintillator material to visible light which generates electron hole pairs in the biased photodiode. The charge carriers are stored in the capacity of the photodiode. By pulsing the gates of a TFT line within the matrix, the charges of all columns are transferred in parallel to the signal outputs.

The detector is divided into two rows of 16 subpanels each, also called *read out groups (ROG),* divided by a midline. The upper and lower part are electrically separated. Each read out group has 128 channels for the detector. The upper groups scan the sensor columns from left to right. The lower groups scan from right to left. The upper groups are transferred first. The upper groups start read out from the upper row. The lower groups start read out from the last row.

4.2 Dysfunctional pixel data

In the literature, dysfunctional pixels are referred to by adjectives including *bad, dead, erratic, stuck, hot, defective, broken and underperforming,* and a variety of conceptions is associated with them. For this data sets we use the *bad pixel map* provided by the detector manufacturer Perkin-Elmer. This is simply a list with all x-y-coordinates of all pixels deemed *underperforming* in regularly taken test images. They use a number of criteria to classify a pixel as *underperforming* based on signal intensities, noise levels, uniformity and lag; see [6] for further details.

Bad pixel maps in X-ray machines are routinely taken after a new detector is installed or an old one is reinstalled after refurbishment. Operators also have the option to take them at times of their convenience. In practice, they usually do so if they feel there “may be something wrong” with the detector.

The data set analysed in this paper comes from a collection of six bad pixel maps taken between June 2013 and January 2014 on a X-ray machine with a *PerkinElmer* digital X-Ray Detector XRD 1621 AN/CN by the Warwick Manufacturing Group. The dimensions of the first two bad pixel maps are 2000 by 2000, but in the third and in the forth bad pixel map they are 2000 by 1600, because the detector was cropped after an excess of bad pixels was detected near the edges. The detector was refurbished between the fourth and the fifth acquisition bringing it back to the initial dimensions. An initial analysis of these data has been performed in [2].

4.3 Spatial analysis

Now we apply the quality assessment methods from Section 3 to the six detector images in our data set. Time point 4 was used for the illustration of the methods in Section 3. For the other time points the functions G, F and K of the pixel process X shows a huge discrepancy to CSR, and still do so, while milder, for the event process Y; not shown here, but similar to what was observed in Figures 4, 5 and 6. However, for each $t = 1, \ldots, 6$ we are again able to define a suitable restricted event process $Y^t_{I-D^t}$ and demonstrate CSR (or nearly CSR in some instances). Results are shown in Figures 12 to 17 below.

In summary, for all time points t in this data set we can show that the algorithms successfully extracts areas of elevated damage D^t by fitting a density and thresholding as explained in Section 3, so that the event process $Y^t_{I-D^t}$ is CSR (or very close to it). Note that the location and shape of D^t closely depend on t.

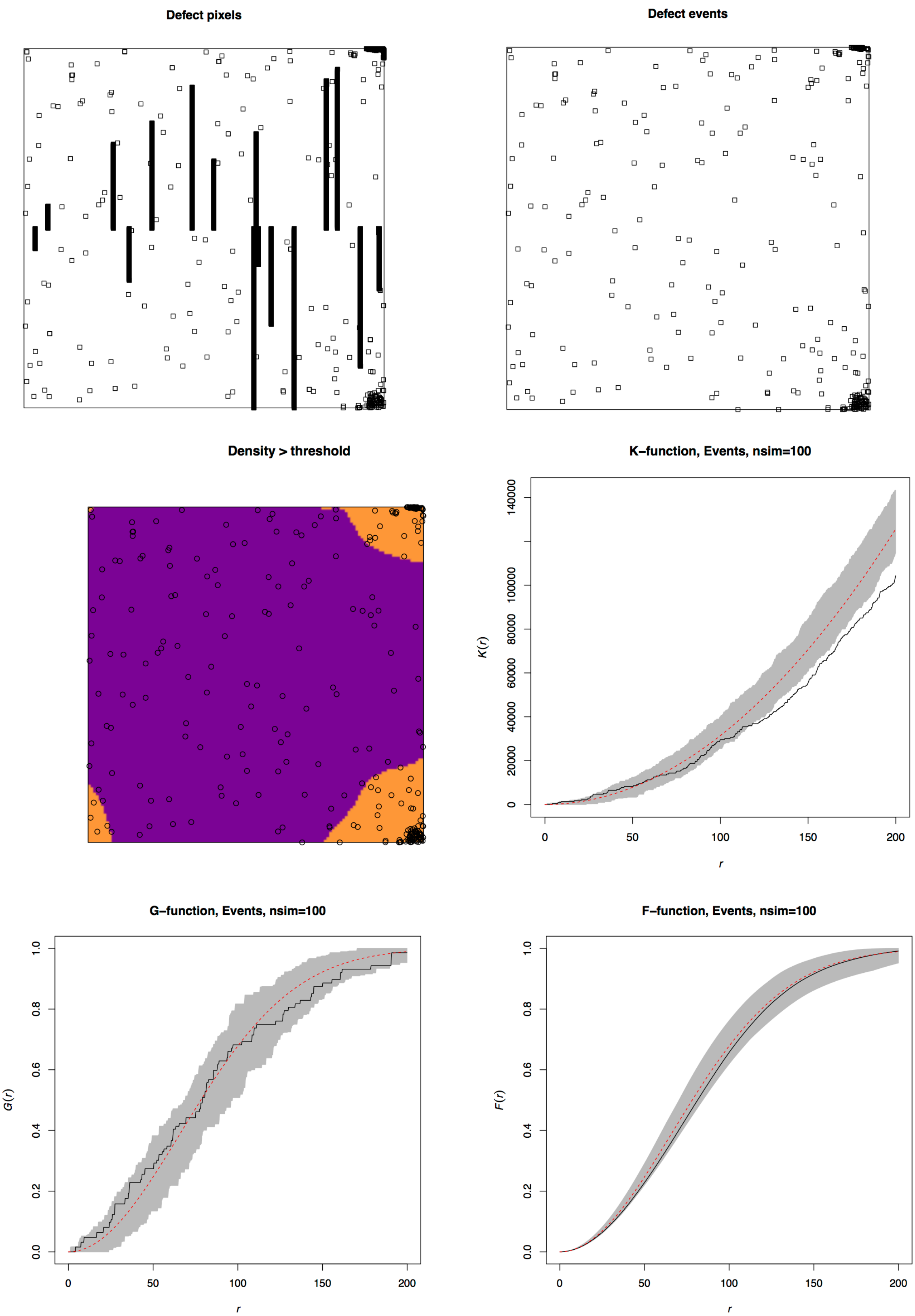


Figure 12: **Detector at time point 1.**

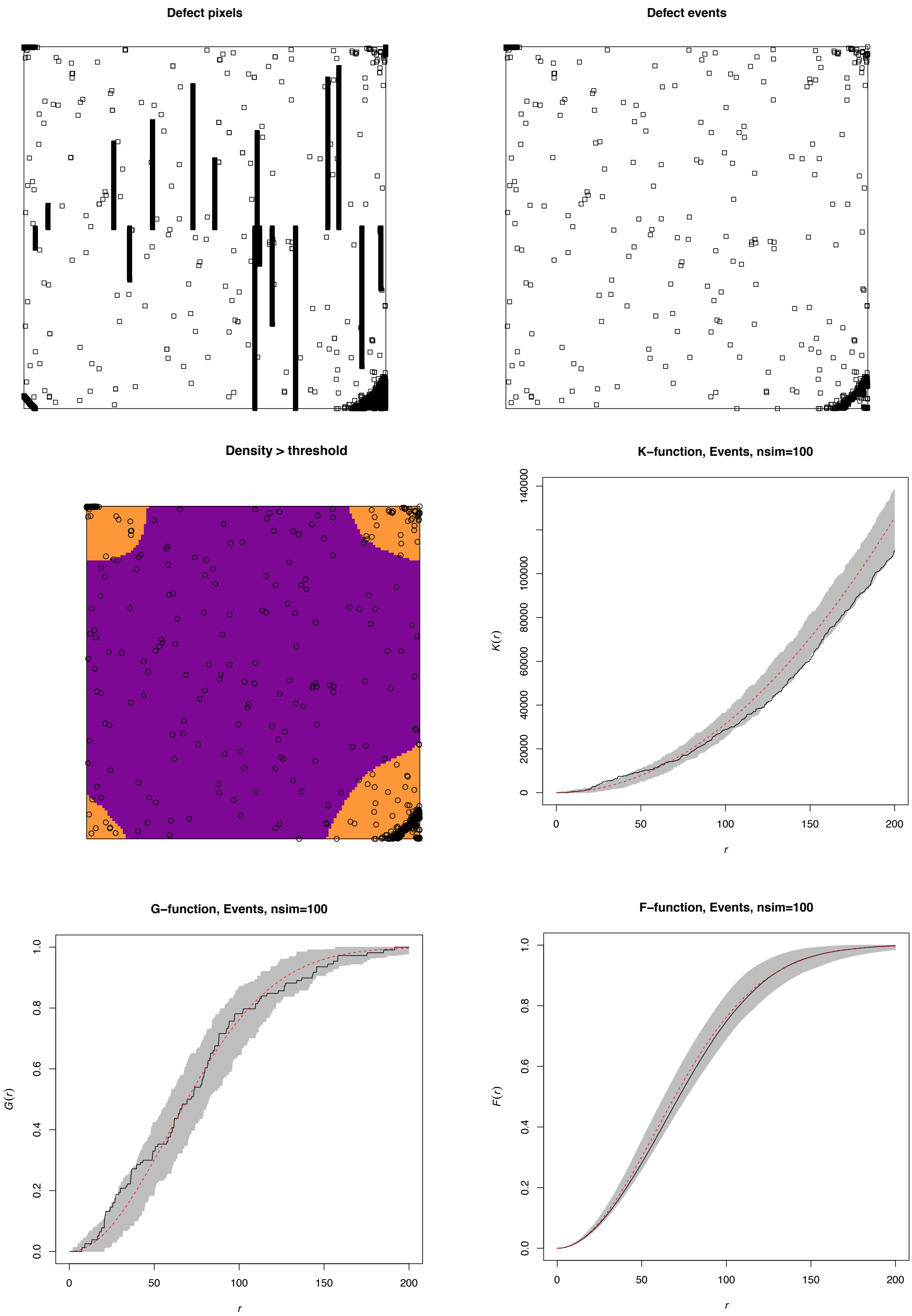


Figure 13: **Detector at time point 2.**

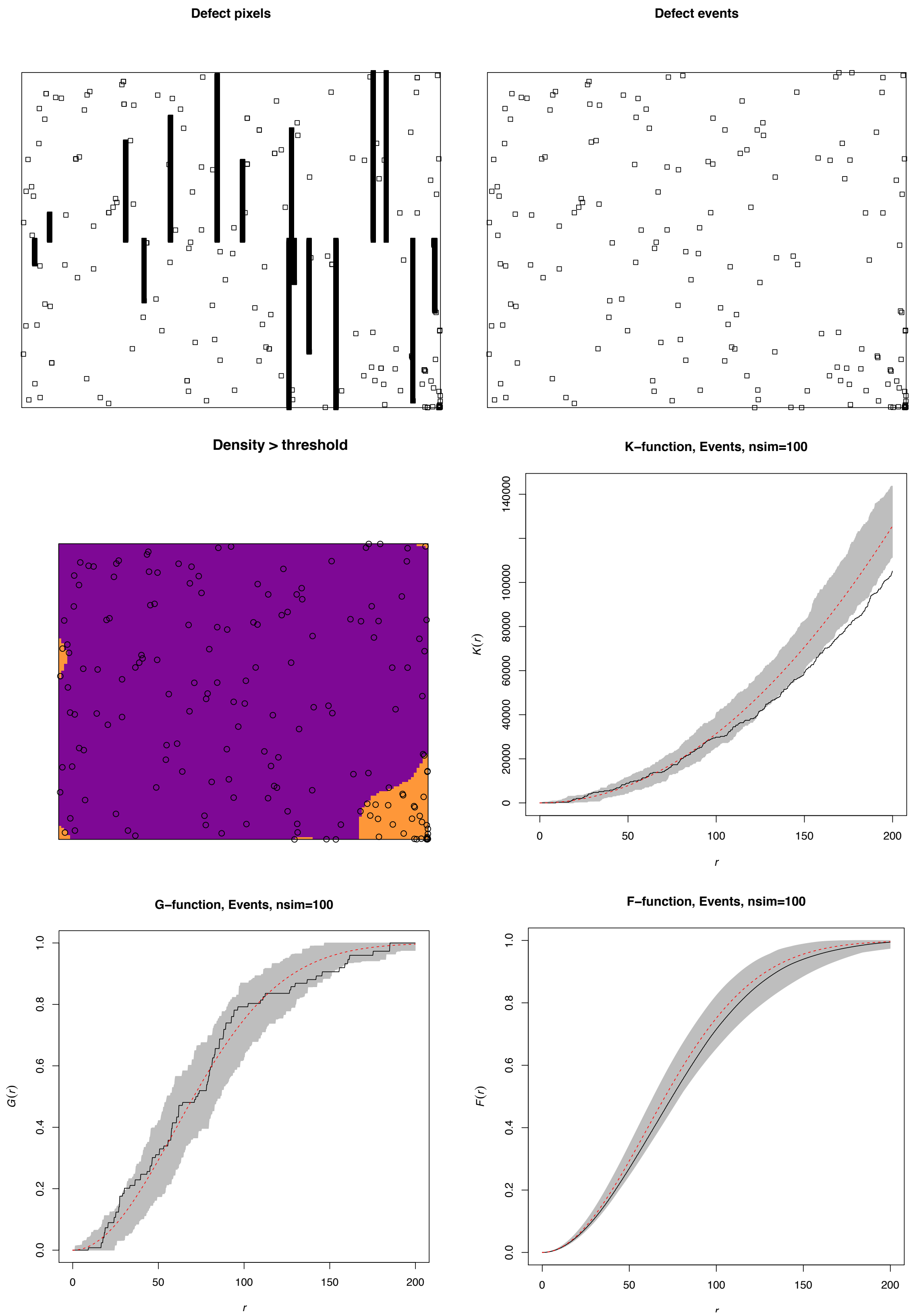


Figure 14: **Detector at time point 3.**

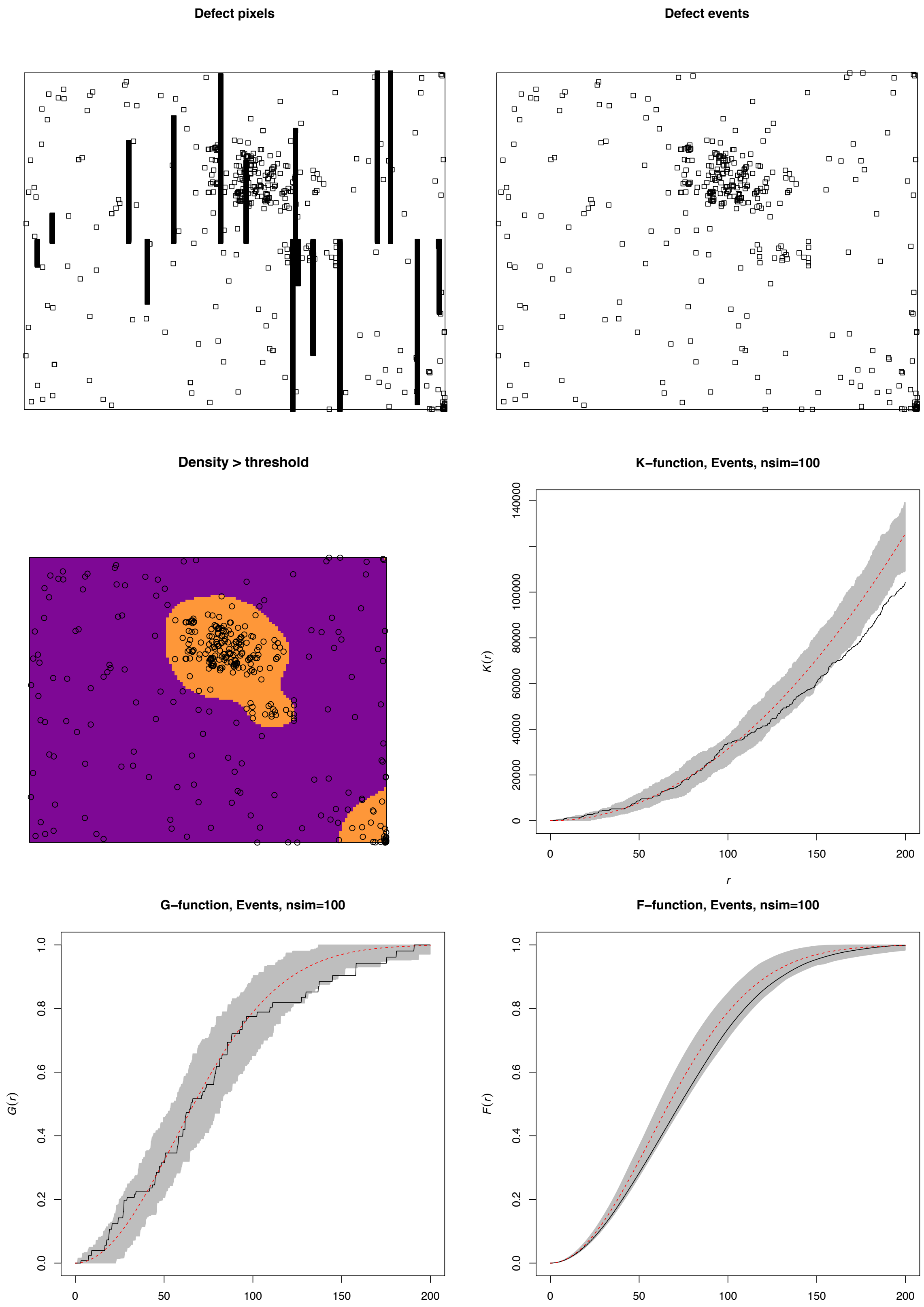


Figure 15: **Detector at time point 4.**

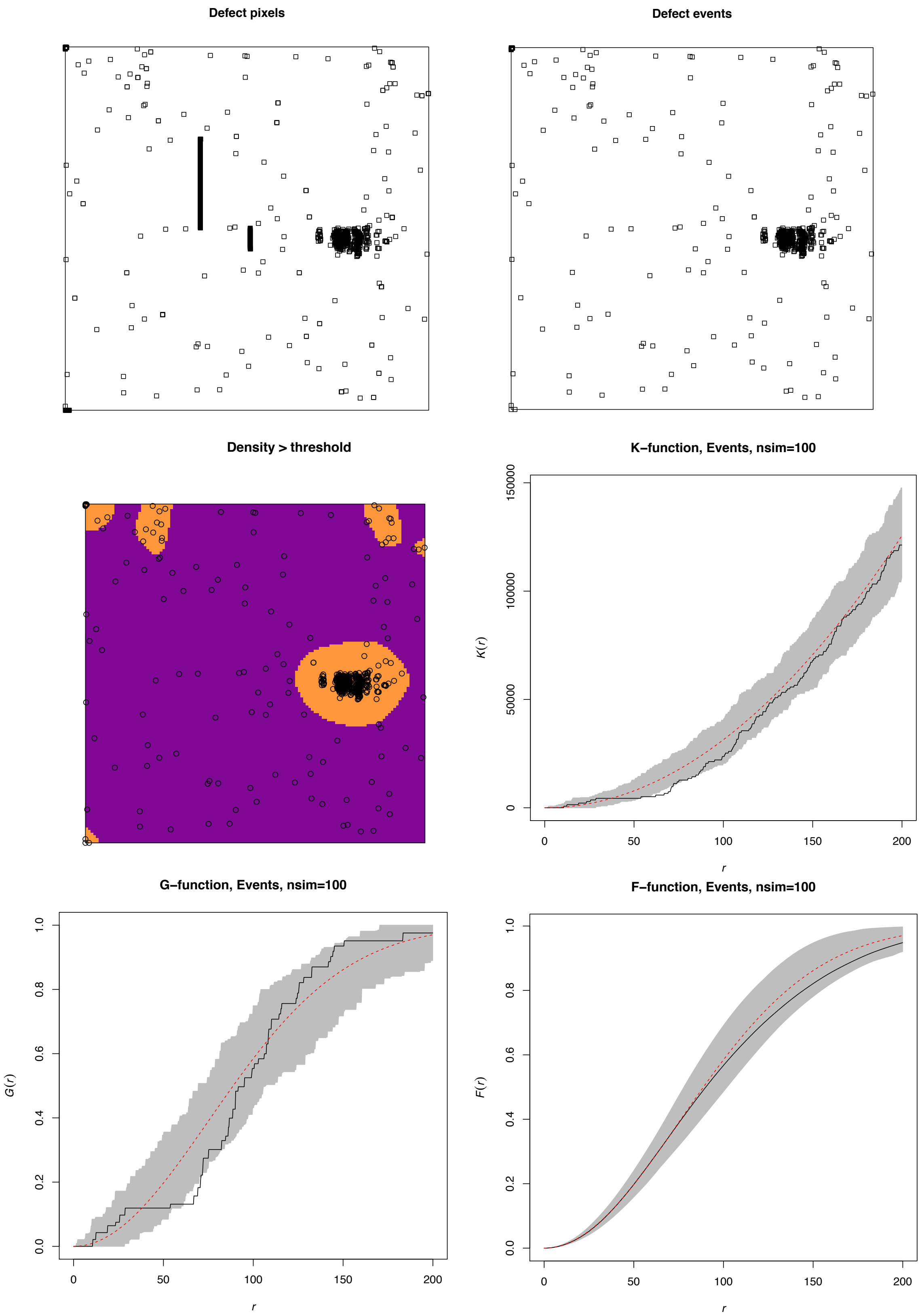


Figure 16: **Detector at time point 5.**

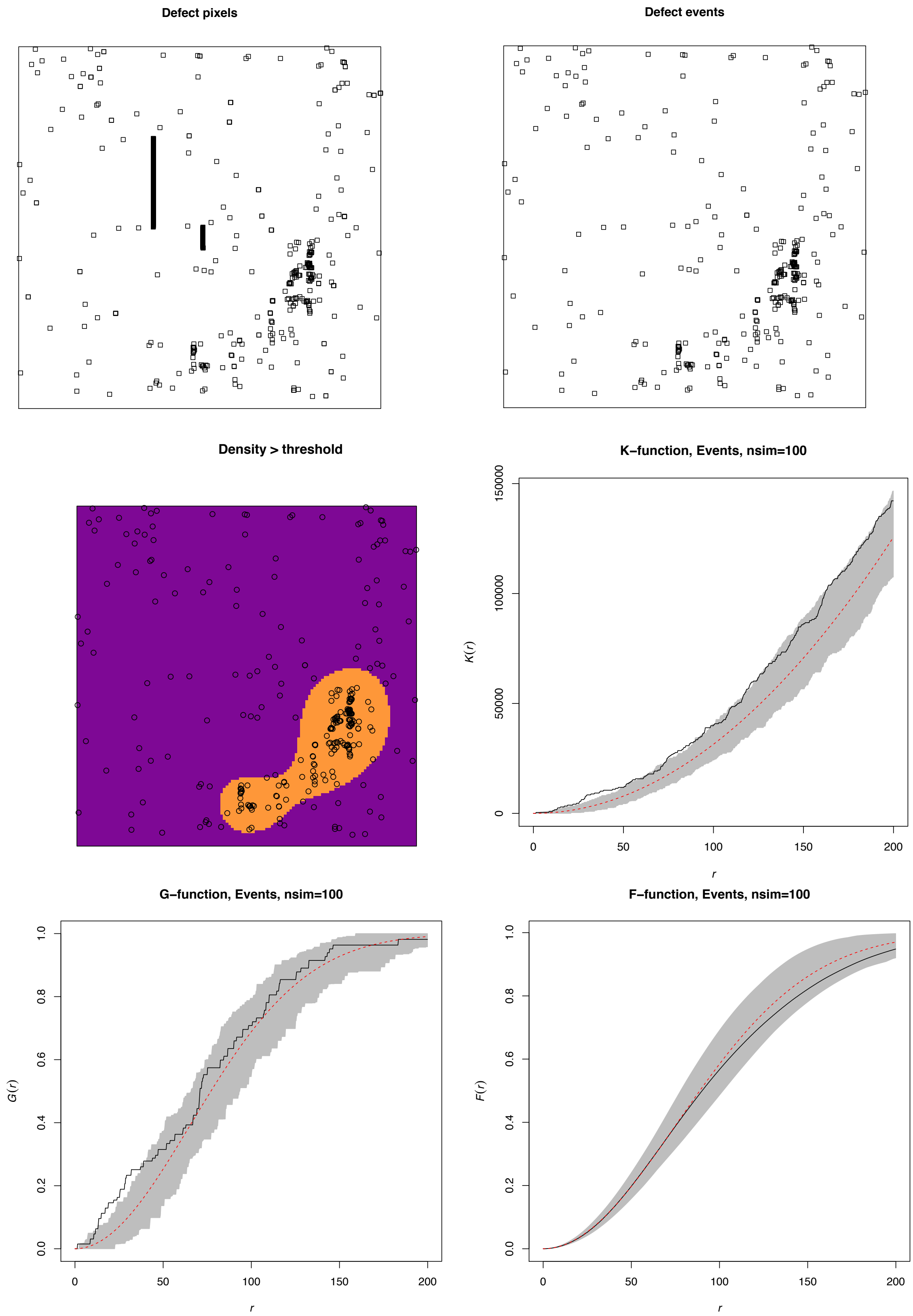


Figure 17: **Detector at time point 6.**

Acknowledgements

The authors wish to acknowledge funding by EPSRC grant EP/K031066/1.

References

[1] Roger S Bivand, Edzer J Pebesma, and Virgilio Gómez-Rubio. *Applied spatial data analysis with R*. Springer, 2008.

[2] Julia Brettschneider, John Thornby, Tom E Nichols, and Wilfried Kendall. Spatial analysis of dead pixels. Technical report, CRiSM Working Paper Series No. 14-24, 2014, 2014.

[3] Angela Cantatore and Pavel Müller. Introduction to computed tomography. Technical report, DTU Mechanical Engineering, 2011.

[4] Maire E. and Withers P. J. Quantitative X-ray tomography. *International Materials Reviews*, 59(1):1–43, 2013.

[5] Carlo Gaetan, Xavier Guyon, and Kevin Bleakley. *Spatial statistics and modeling*, volume 271. Springer, 2010.

[6] PerkinElmer. Reference Manual Digital Imaging, XRD 1621 AN/CN Digital X-Ray Detector.