

DetectorChecker: analyzing patterns of defects in detector screens

Julia A. Brettschneider^{1, 2}, Oscar T. Giles², Wilfrid S. Kendall^{1, 2}, and Tomas Lazauskas²

¹ Department of Statistics, University of Warwick, Coventry, United Kingdom ² The Alan Turing Institute, London, United Kingdom

DOI: [10.21105/joss.02474](https://doi.org/10.21105/joss.02474)

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: [Frederick Boehm](#) ↗

Reviewers:

- [@janfreyberg](#)
- [@craddm](#)

Submitted: 03 July 2020

Published: 22 December 2020

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

DetectorChecker refers to an R package and an associated web application environment *DetectorCheckerWebApp*, intended to help users who need to analyze spatial patterns of defects in images. These images can be *panel-structured*, which is to say, composed of sub-panels arranged in an architecture which the user can specify in the package or in the web application. Primary beneficiaries are intended to be individuals responsible for high-value digital detector screens used in X-ray computerised tomography (XCT), where defects arise due to high radiation flux. More generally the software can be used to analyse defects in other panel-structured arrays, for example solar panels or very large display screens. To maximize accessibility, and to avoid any issues arising from specific software environments, we have created a web application which provides the principal features of the software in standalone form. The web application also affords the possibility of engaging with our team in the analysis of time-evolving defect patterns. To the best of our knowledge, this is the first and presently the only web application and R package facilitating spatial analysis of panel-structured images.

Digital detector screens are crucial high-value components of imaging systems used throughout modern science, medicine, and engineering systems, particularly in XCT. The US [FDA \(2018\)](#) provides information for industry on X-ray imaging devices and lists some common laboratory tests for evaluation of general-use X-ray imaging devices. It also notes the applicable standard for each modality, thus forming the basis for a maintenance schedule. Additionally a scheduled testing framework has been proposed by the Institute of Physics and Engineering in Medicine ([IPEM, 2019](#)). In September 2019 the UK National Health Service (NHS) announced a major investment of £200m to overcome outdated equipment, noting that a significant proportion of CT, MRI and general X-ray equipment more than 10 years old ([NHS, 2019](#)). Thus XCT system quality concerns are very topical.

Yaffe and Rowlands ([1997](#), especially section 3.8) point out that XCT screen quality is linked to system performance. *DetectorChecker* facilitates the inclusion of screen pixel assessment in a testing framework. Note that screen replacement or refurbishment is expensive; regular checks of screen pixels are needed (a) to quantify screen quality and (b) to assess possible *special causes* of defective pixels, using Shewhart's ([1986](#)) terminology from classic quality control. This is best done using spatial statistics, both in order to determine the extent to which spatial patterns of defective pixels can be accounted for by quantifiable independent random variation and also by describing departures from spatial randomness in ways which are suggestive of possible explanations (for example, stress due to screen attachment, or failure at pixel level of data readout). Theoretical spatial statistics methodology is crucial: foundations are discussed in [Chiu et al. \(2013\)](#) while [Baddeley et al. \(2015\)](#) describe the *spatstat* package, an implementation of spatial statistics methods in the R statistical computing environment ([R Core Team, 2019](#)). *DetectorChecker* ([Lazauskas et al., 2020a](#)) is an R package which adapts methods from *spatstat* to the case of panel-structured images, and analyses point patterns arising either from individual defects or from “clumps” of defects (determined in a manner specified by the user). The associated [web application](#) ([Lazauskas et al., 2020b](#)) is based on a self-contained R environment *DetectorCheckerWebApp* together with a *Shiny* ([Chang et al., 2020](#)) gui, implemented and made available *via Azure* at <https://detectorchecker.azurewebsites.net/>. The

web application exposes the basic functionality of the *DetectorChecker* package without the need for users to install R. In particular the web application can be used to define the geometry of the sub-panels of the detector screen (which is to say, the arrangement and size of the component sub-panels), to upload the spatial arrangement of the defective pixels (either directly by means of “bad pixel maps” in XML format or inferred from test images in formats including TIFF), and then to inspect the results using the facilities offered by the package. The software is freely available under MIT licence, accessible *via* the Github repositories in the above references. To the best of our knowledge, there is no comparable package or web application making methods of spatial statistics available for panel-structured image data of arbitrary structure architecture.

Defects are modelled as points in an image rectangle based on overall screen dimensions. The pattern of defects can be modelled using the web application (workflow is summarised in Figure 1).

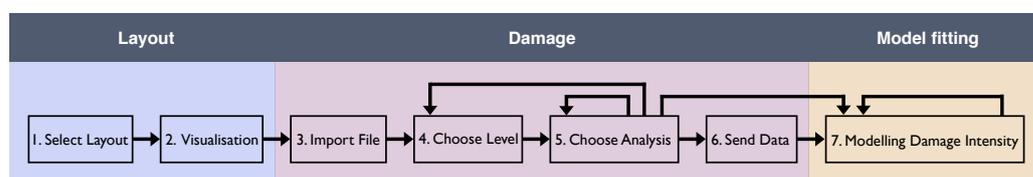


Figure 1: Work flow for DetectorChecker web application. Feedback/skip paths illustrate various options: refocussing attention on subsets of the point pattern (isolated pixels, small clusters, linear clusters, ...); working through various graphical analyses; optionally emailing data to the DetectorChecker team; and statistically fitting a variety of models of damage intensity.

We now discuss selected steps of the workflow using data derived from a Pilatus detector screen and supplied to us by Diamond Lightsource, UK.

A. The user specifies the exact architecture of the sub-panels of the panel-structured image. This can be done either by using a drop-down menu to specify a predetermined option, or by uploading a file giving the specific structure of sub-panels. The data can then be uploaded.

B. Intensity maps can be produced *via* kernel smoothing applied to the point pattern (replacing each defect point by the corresponding translate of a fixed kernel function and then summing). For example, the point pattern in Figure 2(a) yields the intensity map given in Figure 2(b).

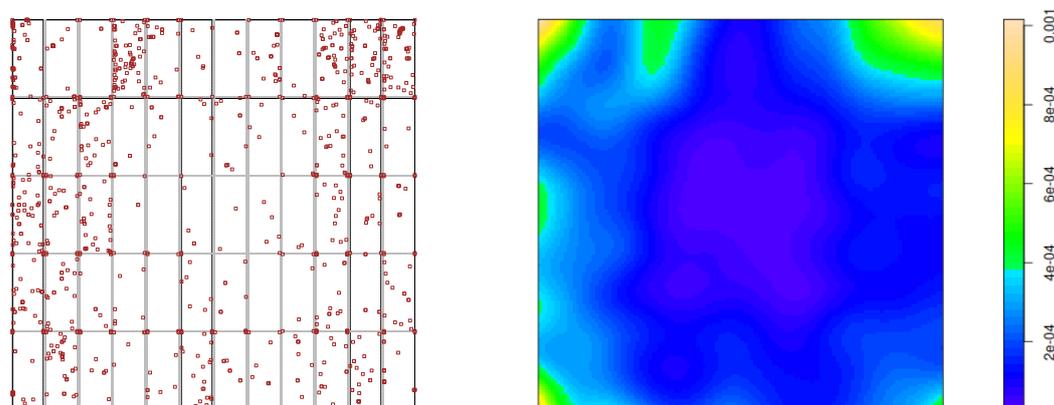


Figure 2: Pilatus detector screen: (a) Example of point pattern of defects. (b) Intensity map resulting from point pattern of defects. The intensity map draws attention to the higher intensity of defects in the corners, which is born out by inspection of the point pattern.

C. Measuring possible departures from *complete spatial randomness* (CSR). CSR is what would be expected if the point pattern was in fact generated by a homogeneous Poisson point process of constant intensity λ , can be assessed using visual inspection of graphs of empirical estimates of F , G and K functions as described in Chiu et al. (2013). It is clear that the point pattern of Figure 2(a) is strongly inhomogeneous and therefore it is not surprising that the corresponding graphical plots indicate clear evidence of deviation from CSR:

- The F function or “empty space function” computes the distribution of the nearest distance to a defect point from a typical location chosen from the image rectangle uniformly at random (and independently of the point pattern). If the point pattern did in fact satisfy CSR then one could consider an empirical estimate of the F function to be a random perturbation of the theoretical F function under CSR, namely $F_{\text{pois}}(r) = 1 - \exp(-\lambda\pi r^2)$. Figure 3(a) graphs different variants of \hat{F} , accounting in various ways for edge-effects. Note the clear deviation of the \hat{F} empirical estimates from what would be expected under CSR, namely the theoretical F_{pois} .
- The G function computes the distribution of nearest-neighbour distances between defect points; if the point pattern did in fact satisfy CSR then one could view an empirical estimate of the G function as a random perturbation of the theoretical G function under CSR, namely (by a conditional probability argument) $G_{\text{pois}}(r) = 1 - \exp(-\lambda\pi r^2)$ (actually equal to $F_{\text{pois}}(r)$). See Figure 3(b), and note the clear deviation from the theoretical G_{pois} of the \hat{G} empirical estimates (which again account in various ways for edge-effects), hence again suggesting deviation from CSR.

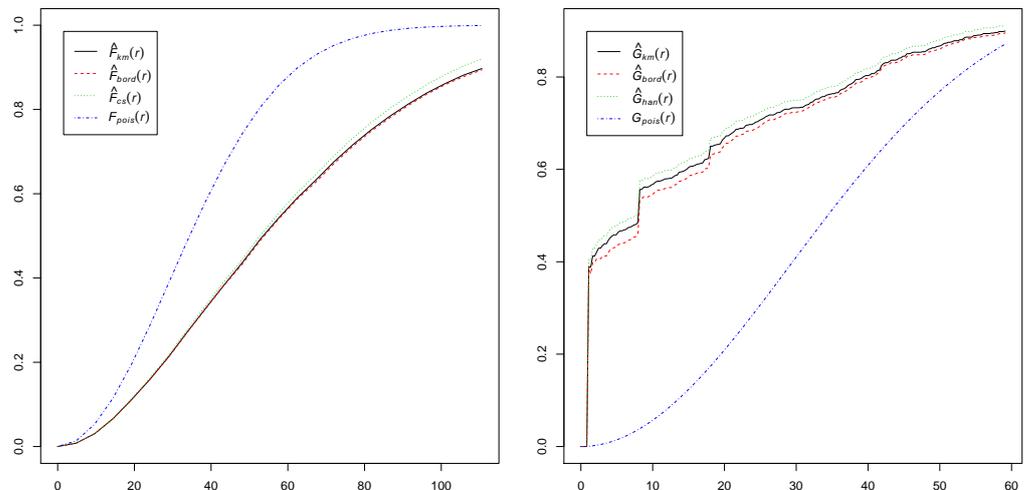


Figure 3: Pilatus detector screen: (a) F plot resulting from point pattern of defects. (b) G plot resulting from point pattern of defects. The graphs contrast the theoretical curve arising from CSR (blue dotted curve) with several empirical curves involving different edge corrections. The different edge-corrected empirical curves agree with each other but indicate clear divergence from the CSR curve. Here and in the following figures the graphs correspond to output from the application: graph axes have not been harmonized.

- The K function (Ripley’s K function) computes the mean number of defect points within a distance r of a typical defect point, viewed as a function of r ; if the point pattern did in fact satisfy CSR then one could view an empirical estimate of the K function as a random perturbation of the theoretical K function under CSR, namely $K_{\text{pois}}(r) = \pi r^2$. See Figure 4(a), and note the deviation from K_{pois} of the \hat{K} empirical

estimates (once more accounting for edge-effects in different ways), especially at short distances, once more suggesting deviation from CSR. Note that for geometrical reasons the \hat{K} empirical estimates will exhibit substantially greater variation at large distances; it is therefore appropriate to confine attention to the left-hand third of the x -axis. The excess over the theoretical K_{pois} at short distances, particularly for the estimate \hat{K}_{iso} , indicates that defects are more clustered than would be expected from CSR.

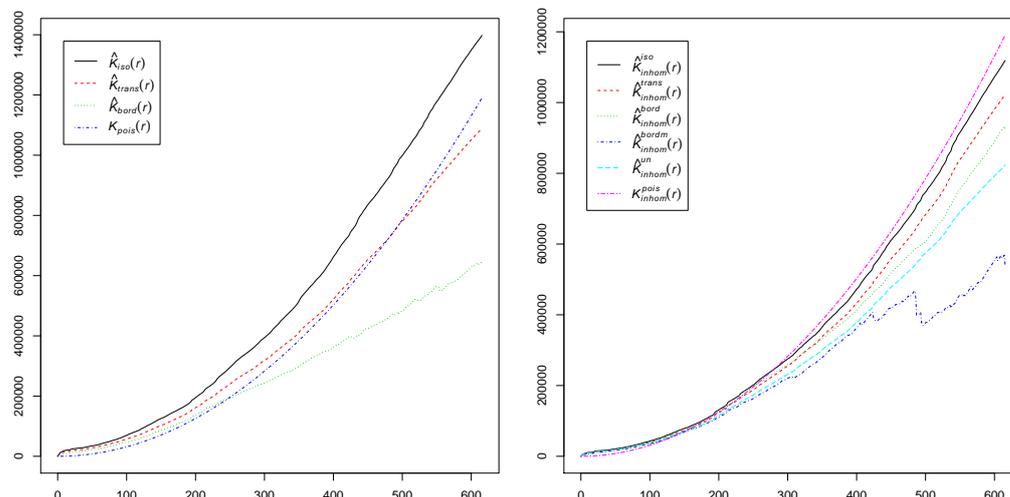


Figure 4: Pilatus detector screen: (a) K plot resulting from point pattern of defects. (b) K plot resulting from point pattern of defects, corrected for inhomogeneity. For geometrical reasons it is appropriate to focus attention on small distances. There is more variation between different edge-corrections of empirical curves than for F and G curves. Empirical curves are closer at short distances (200 pixels or less) to the theoretical curve based on CSR (left panel, blue dotted curve) but still exhibit some discrepancy hinting at possibly greater clustering relative to CSR. However all curves agree closely for short distances in the right panel, in which a correction has been made for inhomogeneity (which has already been noted when considering the intensity map). This suggests that an inhomogeneous Poisson process provides a good fit for the data.

- Plots are also available which take account of inhomogeneity and compare these estimates to theoretical functions computed for inhomogeneous Poisson point processes: Figure 4(b) gives an example of this in the case of the K function. The plots of the \hat{K}_{inhom} empirical inhomogeneity-adjusted estimates agree much more closely with the theoretical $K_{\text{inhom}}^{\text{pois}}$ function at short distances, supporting the hypothesis that the pattern of defects is what might be expected to arise from an *inhomogeneous* Poisson process of defects.

D. Finally the relationship of the defect points to sub-panel boundaries can be studied by means of various logistic regression options, which assess whether damage intensity appears to depend on distance from the centre of the image or horizontal or vertical distance from sub-panel edges. When this data set is modelled in terms of Euclidean distance from the centre, the web application reports substantial evidence for positive dependence of defect intensity on distance from the centre (see the highly significant coefficient for `as.vector(dist)` in the following web application output), conforming with the visual impression given by Figure 2(a), and further explaining the nature of the spatial inhomogeneity indicated by Figure 4. In fact this positive dependence reflects manufacturing details of this particular screen design: Diamond reports that Pilatus detector screen panels are tested before installation, and better panels are placed in the centre of the structured display.

Here follows output from the web application after performing the above logistic regression:

```
Call:
glm(formula = as.vector(pix_matrix) ~ as.vector(dist),
family = binomial(link = logit))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.0262  -0.0211  -0.0193  -0.0171   4.3297

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)    -9.392e+00  9.628e-02  -97.545  <2e-16 ***
as.vector(dist)  8.021e-04  8.726e-05   9.192  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 22261  on 6224000  degrees of freedom
Residual deviance: 22173  on 6223999  degrees of freedom
AIC: 22177

Number of Fisher Scoring iterations: 11
```

The web application also provides for further graphical options, such as the study of direction from a typical defect point to its nearest neighbour within the relevant sub-panel, analysis at the level of “events” (appropriately defined grouping of clumps of defect pixels) rather than individual defect points, and exclusion of regions of the image rectangle for which the defect intensity is clearly different (this often arises in XCT, where corners of the image exhibit high defect intensity, presumably deriving from mechanical stress due to supports of the screen).

An extended example of use of the R package, paralleled by corresponding use of the web application, is available as a vignette in the Github repository [DetectorChecker](#).

The R package and web application together offer significant opportunities to address interesting and important challenges for the data analysis of defective pixel patterns. The web application offers the possibility of uploading users’ data to a data repository, thus permitting the possibility of organizing cooperative statistical investigations comparing patterns across different machines and different modes of usage. In particular we envisage its use to collect time sequences of images, to permit statistical investigation by the Warwick team of deterioration over time, using latent Markov models of the life and death of defective pixels which are currently being developed. Such analysis requires sustained and regular monitoring of a diversity of screens from various devices, together with recording of relevant metadata such as detector usage. Interested users are encouraged to make contact to discuss these possibilities, which will permit evidence-based analysis to support decisions on refurbishment and/or replacement strategies.

Acknowledgements

We gratefully acknowledge support from the UK EPSRC (grant EP/K031066/1) and the Alan Turing Institute (under the EPSRC grant EP/N510129/1) during this project.

We also wish to thank Martin O'Reilly (The Alan Turing Institute), Nicola Tartoni and Ian Horswell (Diamond Lightsource, UK) for guidance on detector types and sample data sets, and Tristan Lowe (Henry Moseley X-ray Imaging Facility, University of Manchester) and Martin Turner (ITS Research IT, University of Manchester) for discussions and feedback.

References

- Baddeley, A., Rubak, E., & Turner, R. (2015). *Spatial point patterns: Methodology and applications with R*. Chapman; Hall/CRC. <https://doi.org/10.1002/bimj.201600178>
- Chang, W., Cheng, J., Allaire, J., Xie, Y., & McPherson, J. (2020). *Shiny: Web application framework for r*. <https://CRAN.R-project.org/package=shiny>
- Chiu, S. N., Stoyan, D., Kendall, W. S., & Mecke, J. (2013). *Stochastic Geometry and Its Applications* (p. 518). John Wiley; Sons. <https://doi.org/10.1002/9781118658222>
- FDA. (2018). *Information for industry: X-ray imaging devices*. <https://www.fda.gov/radiation-emitting-products/medical-x-ray-imaging/information-industry-x-ray-imaging-devices>
- IPEM. (2019). *Quality assurance*. <https://www.radiologycafe.com/radiology-trainees/frcr-physics-notes/quality-assurance>
- Lazauskas, T., Giles, O. T., Kendall, W. S., & Brettschneider, J. A. (2020a). <https://github.com/alan-turing-institute/DetectorChecker> (Version 1.0.6) [Computer software].
- Lazauskas, T., Giles, O. T., Kendall, W. S., & Brettschneider, J. A. (2020b). <https://github.com/alan-turing-institute/DetectorCheckerWebApp> (Version 1.0.7) [Computer software].
- NHS. (2019). *Transforming imaging services in england: A national strategy for imaging networks*. https://improvement.nhs.uk/documents/6119/Transforming_imaging_services.pdf
- R Core Team. (2019). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. <https://www.R-project.org>
- Shewhart, W. A. (1986). *Statistical method from the viewpoint of quality control* (Second, p. x+155). Dover Publications, Inc., New York. ISBN: 0-486-65232-7
- Yaffe, M. J., & Rowlands, J. A. (1997). X-ray detectors for digital radiography. *Physics in Medicine and Biology*, 42(1), 1–39. <https://doi.org/10.1088/0031-9155/42/1/001>