

# Unbiased Population Size Estimation on Still Gigapixel Images

Sociological Methods &amp; Research

1-22

© The Author(s) 2018

Article reuse guidelines:

[sagepub.com/journals-permissions](https://sagepub.com/journals-permissions)

DOI: 10.1177/0049124118799373

[journals.sagepub.com/home/smr](https://journals.sagepub.com/home/smr)

Marcos Cruz<sup>1</sup>  and Javier González-Villa<sup>1</sup> 

## Abstract

Population sizing is essential in sociology and in various other real-life applications. Gigapixel cameras can provide high-resolution images of an entire population in many cases. However, exhaustive manual counting is tedious, slow, and difficult to verify, whereas current computer vision methods are biased and known to fail for large populations. A design unbiased method based on geometric sampling has recently been proposed. It typically requires only between 50 and 100 manual counts to achieve relative standard errors of 5–10 percent irrespective of population size. However, the large perspective effect introduced by gigapixel images may boost the relative standard error to 30–40 percent. Here, we show that projecting the sampling grid from a map onto the gigapixel image using the camera projection neutralizes the variance due to perspective effects and restores the relative standard errors back into the 5–10 percent range. The method is tested on six simulated images. A detailed step-by-step illustration is provided with a real image of a 30,000 people crowd.

## Keywords

crowd size, geometric sampling, demonstration, design unbiased method, population size, political rally, gigapixel image

---

<sup>1</sup> Universidad de Cantabria, Santander, Cantabria, Spain

## Corresponding Author:

Marcos Cruz, Universidad de Cantabria, Av. Los Castros 48, Santander, Cantabria 39005, Spain.

Email: [marcos.cruz@unican.es](mailto:marcos.cruz@unican.es)

## Introduction

A population is a finite set of separate items or “particles” of interest. Determining the total number of particles in the population, that is, the population size,  $N$ , is important in many real-world problems. Human crowd sizing is relevant to sociology (e.g., demonstrations, political rallies, concerts, marathons), whereas herd, flock, or swarm sizes are useful in ecology.

The traditional density method (Jacobs 1967; Watson and Yip 2011) is still widely used for crowd sizing. However, it has high and unpredictable errors and is often not verifiable. Frequently, size estimates differ widely among convention organizers, media, and police.

High-resolution photography has made it possible to image the entire population or at least an important sample of it (Lynch, Alderman, and Hobday 2015), so that the particles are unambiguously distinguishable for counting. However, for population sizes above a few thousands, manual counting in still images is too laborious, slow, observer dependent, and difficult to verify.

A number of computer vision methods have been proposed in the particular case of human crowd sizing on still images (see, for instance, Botta, Moat, and Preis 2015; Idrees et al. 2013; Lempitsky and Zisserman 2010; Rodriguez et al. 2011; Zhang et al. 2015). Unfortunately, they are unable to handle crowd sizes above a few thousands and there is still a 10-fold improvement to be made before reaching human-based performance (Zhang et al. 2016). The application of these methods to ecology needs specific remodeling for the particular target population (see, for instance, Descamps et al. 2011). It is well-known that estimators have two sources of error, namely, variance and bias. Automatic computer vision methods are biased since their error is due to systematic errors (e.g., false detections and false negatives), which do not have 0 expected value and are unpredictable.

Recently, Cruz, Gómez, and Cruz-Orive (2015) proposed an unbiased population size estimation method (hereafter *CountEm* method, with reference to the free software available at <http://countem.unican.es>) that can be applied to any kind of particle. It was adapted from well-known principles of geometric sampling for stereology, which are widely used in many disciplines (see Baddeley and Jensen 2005; Cruz-Orive 2017; Howard and Reed 2005, and references therein). The main tool is systematic sampling with a uniform random (UR) grid of quadrats. The standard choice is a square grid of square quadrats defined by two parameters, namely, the separation,  $T$ , between quadrat centers and the quadrat side length,  $t$ , ( $0 < t \leq T < \infty$ ) (see Figure 1). The total number,  $N$ , of particles in a given image is estimated



**Figure 1.** (a) A portion of the grid of quadrats proposed in Cruz et al. (2015) for systematic sampling. The sampling period is  $T^2/t^2$ . (b) Magnified version of the quadrat marked in (c). Only the two marked heads should be counted, considering heads as counting units and applying the forbidden line counting rule (Cruz et al. 2015). (c) Low-resolution version of a gigapixel picture by Adriano Morán (93 metros) of a demonstration at Puerta del Sol in Madrid. The high-resolution gigapixel image can be better appreciated in Morán (2015). A grid of quadrats of the type shown in (a), with  $T = 4,000$  and  $t = 200$  pixels, has been superimposed with a tilt of  $60^\circ$ .

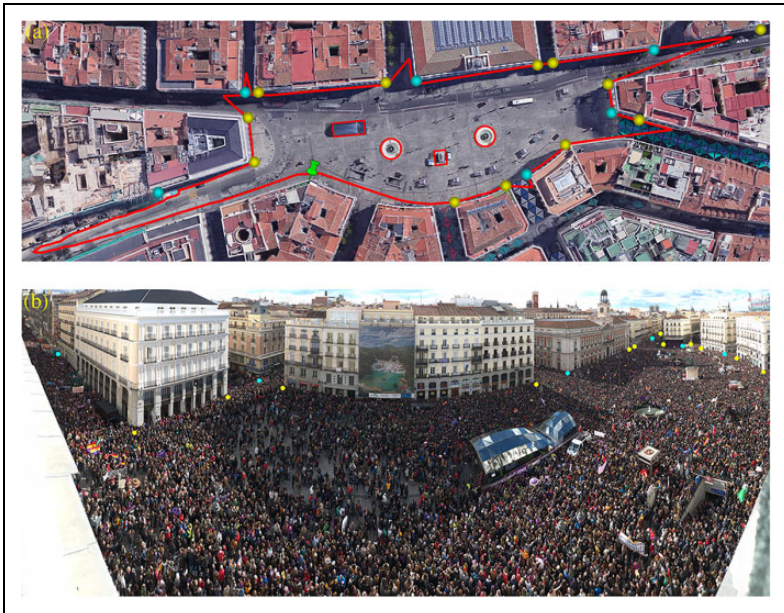
by the total number,  $Q$ , of units captured by the quadrats  $\times$  the sampling period  $T^2/t^2$ . The estimator  $\hat{N}$  is unbiased, which means that for any pair  $(t, T)$ , the mean of the error  $\hat{N} - N$  over all potential UR superimpositions of the grid is equal to 0, as long as the sampling units are distinguishable for counting. Thus, unbiasedness is a mathematical property, which renders empirical validation unnecessary in the absence of observation artifacts. Error variance predictors from a single sample, however, are generally not unbiased approximations and require testing.

The *CountEm* estimator was tested by Cruz et al. (2015) for two annotated crowd images, namely, the spectators image which can be downloaded at <http://countem.unican.es> and fig. 7b of Idrees et al. (2013). The number of annotated humans in these images was 1,220 and 4,633, respectively. Monte

Carlo resampling showed that manually counting about 50(100) individuals sampled in about 20 nonempty quadrats can yield relative standard errors of about 8 percent (5 percent). With such small sample sizes, *CountEm* is an attractive alternative to automatic methods, which are hitherto biased to unknown degrees. The resampling experiments were mainly intended to test the performance of a new error variance prediction formula (*Cavalieri* estimator,  $\text{var}_{\text{Cav}}(\hat{N})$ , eq. 3 of Cruz et al. 2015) which is also tested for gigapixel images in the present article.

Here, the *CountEm* method is implemented and tested on a  $51,350 \times 21,078$  pixels gigapixel image of a demonstration at *Puerta del Sol*, Madrid, taken in January 2015 by Adriano Morán (<http://93metros.com>). The full resolution picture can be seen in (Morán, 2015), whereas a low-resolution version is shown in Figure 1c. Estimating the size of the crowd pictured in this image is challenging since it is about an order of magnitude higher than the crowd sizes considered in computer vision studies (Idrees et al. 2013; Lempitsky and Zisserman 2010; Rodriguez et al. 2011; Zhang et al. 2015). Satellite or aircraft images do usually have insufficient resolution to allow unambiguous counting. Gigapixel pictures yield high resolutions, but they are often ground based, producing inhomogeneous population patterns due to perspective effects. The *CountEm* method is unbiased irrespective of population size and pattern, but the sampling variance depends on the distribution of quadrat counts. The huge perspective artifacts shown for instance in Figure 1c may significantly increase the variance. Therefore, we propose to generate a UR standard grid of quadrats on the map of the area where the picture has been taken (see Figures 2a and 3a) and transform it using the camera projection (Figure 3c). The grid transformation preserves unbiasedness. Hereafter the new implementation is called *CountEm2*. Automatic resampling on a simulated point cloud (Figure 3c) resembling the real image Figure 1c revealed that *CountEm2* could reduce the relative standard error from about 30 percent (if the method in Cruz et al. 2015 is directly applied) down to 5–10 percent.

The article is organized as follows: The *CountEm* and *CountEm2* methods are outlined in Material and Method section. The projection formulas and the camera parameter estimation are explained in Cylindrical Projection: Cylindrical Coordinates section. In Crowd Coordinate Simulation section, we describe how to simulate the point clouds shown in Figure 3. The results of the Monte Carlo resampling analysis are presented in Outline of the CountEm Method section. A robustness test is performed in Population Sizing With the Gigapixel Picture section, in order to assess the impact of possible projection inaccuracies. Population sizing with the real gigapixel



**Figure 2.** (a) Aerial view of the Puerta del Sol, Madrid (PNOA, Instituto Geográfico Nacional). Reference points for parameter estimation are marked in yellow and cyan. (b) Same image as in Figure 1c with the reference points of (a).

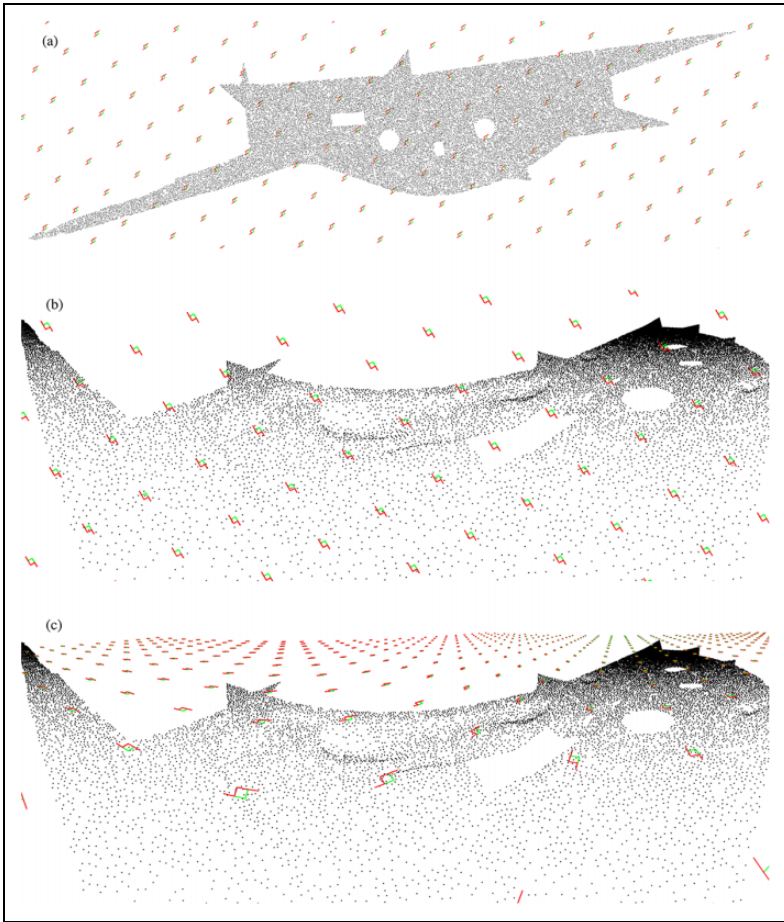
picture is illustrated in Population Sizing With the Gigapixel Picture section. Finally, the conclusions are given in Summary and Conclusion section.

## Material and Method

### Outline of the CountEm Method

The main steps of the standard *CountEm* method (Cruz et al. 2015) are:

1. Define a sampling grid by choosing parameters  $t$  and  $T$  (see Figure 1a). Two practical criteria were given in Cruz et al. (2015) to design an efficient grid. First, aim at a sample size  $Q$  of about 100 particles and second, aim at counting no more than 4 or 5 particles per quadrat. In images with weak perspective effects, this typically yields a coefficient of error below 10 percent irrespective of population size  $N$ . Optionally, the grid might be tilted at will a given, fixed angle in order to avoid undesirable alignments of the population pattern and



**Figure 3.** (a) Simulation of crowd positions (1 dot = 1 person) on a map based on Figure 2a. A standard grid of systematic quadrats has been superimposed on it (Traditional *CountEm* method). (b) Unwrapped cylindrical projection of the crowd in (a) with a traditional *CountEm* grid. (c) Same crowd projection as in (b) with unwrapped cylindrical projection of the grid in (a) (novel *CountEm2* method proposed in this article).

the grid rows. An alignment does not affect unbiasedness, but it might induce higher error variance. For instance, a whole row of spectators can be included or excluded of the sample depending on the particular grid position (see fig. 11 of Gundersen et al. 1999).

2. Superimpose the grid uniformly at random on the crowd image, for example, Figure 1c.
3. Manually count the total number,  $Q$ , of particles captured by the quadrats and multiply by the sampling period  $T^2/t^2$ :

$$\hat{N} = \frac{T^2}{t^2} \cdot Q. \quad (1)$$

Note that the forbidden line rule should be used to avoid edge effects leading to biased counting (Baddeley 1999; Baddeley and Jensen 2005; Cruz et al. 2015; Gundersen 1977). A particle is counted in a quadrat only if it intersects the quadrat but it does not hit the extended forbidden line of the quadrat (in red in Figure 1b).

4. Estimate the sampling variance using the Cavalieri estimator proposed in eq. 3 of Cruz et al. (2015).

### Outline of the CountEm2 Method

The basic steps of the proposed *CountEm2* population size estimation method are as follows:

1. Identify the area occupied by the target population in a map, for example, Figure 2a. Note that we outline the reference area for simulation purposes and it is not necessary otherwise.
2. Apply standard *CountEm* steps (1) and (2) on the map, for example, Figure 3a. We recommend to transform longitude and latitude coordinates into meters choosing an arbitrary neighboring origin and to obtain the corresponding sea-level elevations if available.
3. Project the grid onto the gigapixel picture using the camera projection parameters as explained in Cylindrical Projection: Cylindrical Coordinates section, for example, Figure 3c or Figure 7.
4. Apply standard *CountEm* steps (3) and (4) to the gigapixel image with the superimposed projected grid.

### Cylindrical Projection: Cylindrical Coordinates

Cylindrical projection allows a large field of view and preserves vertical lines. Therefore, it is often the preferred option for gigapixel images such as Figure 1c. Our aim is to project a grid of quadrats from real-world 3D coordinates of a map onto the gigapixel image using the relevant cylindrical

camera projection parameters. Hence, we need to know how to project an arbitrary point in real space,  $X \in \mathbb{R}^3$  with known coordinates  $X = (x_1, x_2, x_3)$  in arbitrary distance units, into a point  $G$  in the gigapixel image, with coordinates  $G = (g_1, g_2)$ . We chose the  $x_1$  and  $x_2$  axes defining a horizontal ground plane with a vertical  $x_3$ -axis and the origin at the location of the observer  $O \in \mathbb{R}^3$ . The  $g_1$  and  $g_2$  axes in the image follow the image analysis convention, with the  $g_1$ -axis increasing from left to right, the  $g_2$ -axis increasing downward, and the origin at the upper-left corner of the image.

If the camera projection parameters are unknown, then they can be estimated using simple pinhole camera equations and applying the following steps.

First, consider a right circular cylinder of radius  $r > 0$  with symmetry axis along the  $x_3$ -axis. The point  $X$ , with cylindrical coordinates  $X = (R, \phi, x_3)$ , is projected by a ray emanating from  $O$  into a point  $C = (r, \phi, z)$  on the cylinder surface. It is easy to show that

$$\begin{aligned} R &= \sqrt{x_1^2 + x_2^2} \\ \phi &= \text{atan2}(-x_2, -x_1) \\ z &= -\frac{r}{R}x_3. \end{aligned} \quad (2)$$

Note that there are two possible intersections of the ray with the cylinder. In the latter two equations, the intersection corresponding to the minus signs is chosen in order to obtain the desired image orientation.

Unwrapping the cylinder, the dimensionality is reduced from 3 to 2, and  $C \in \mathbb{R}^3$  is transformed into  $P \in \mathbb{R}^2$  with coordinates  $P = (p_1, p_2)$ , namely:

$$\begin{aligned} p_1 &= r\phi \\ p_2 &= -\frac{r}{R}x_3. \end{aligned} \quad (3)$$

Since the origin is arbitrary, we apply a shift  $\Delta = (\Delta_1, \Delta_2)$  to  $P$  in order to match the photo coordinates. Thus, each point  $X$  is mapped into a point  $G = P + \Delta = (g_1, g_2)$  in the image, where

$$\begin{aligned} g_1 &= r\phi + \Delta_1 \\ g_2 &= -\frac{r}{R}x_3 + \Delta_2, \end{aligned} \quad (4)$$

with  $\phi \in (-\pi, \pi]$ . The parameter vector  $\theta = (\Delta_1, \Delta_2, r)$  has to be estimated in order to perform the desired projection of the quadrat grid as in Figure 3c.



In future applications, it should be technically possible to read the projection parameters directly from the camera. The camera information was unfortunately not available for the gigapixel image Figure 1c. Therefore, the required parameters had to be estimated as we describe below.

### *Cylindrical Projection: Parameter Estimation*

Several reference points are needed in order to estimate the parameter vector,  $\theta$ . Each reference point  $G = (g_1, g_2)$  in the gigapixel image has to be identified and matched with its corresponding real-space coordinates  $X = (x_1, x_2, x_3)$ .

We used the 18 points represented with yellow and cyan dots in Figure 2. The longitude and latitude coordinates of a point  $X$  can be transformed into 3D coordinates  $(x_1, x_2, x_3)$  in meters choosing an arbitrary origin and using, for instance, the Python Geocoding Toolbox and Google Earth Pro elevation data or similar tools. Their coordinates in degrees, sea-level elevations in meters, and corresponding  $(g_1, g_2)$  photo coordinates in pixels are listed in Table 1. The point number 0 corresponds to the observer location  $O$  which was determined with Google Earth Pro and is marked with a green pushpin icon in Figure 2a.

An overdetermined system can be obtained from the foregoing equations, namely:

$$\begin{aligned} \text{atan2}(-x_2, -x_1)r + \Delta_1 &= g_1 \\ \frac{-x_3}{\sqrt{x_1^2 + x_2^2}}r + \Delta_2 &= g_2, \end{aligned} \quad (5)$$

which can be written in matrix notation as  $\mathbf{A}\theta = \mathbf{b}$ .  $\mathbf{A}$  is the coefficient matrix and  $\mathbf{b}$  the independent term vector. An approximate solution to the overdetermined system can be found using ordinary least squares. Solving the system with the reference points of Table 1 and choosing the origin at  $O$ , we obtain:

$$\begin{aligned} r &= 1,428.3 \text{ pixels} \\ \Delta_1 &= 2,587.9 \text{ pixels} \\ \Delta_2 &= 1,62.3 \text{ pixels.} \end{aligned} \quad (6)$$

Note that for convenience, the distance units for  $x_1, x_2$ , and  $x_3$  are meters, whereas for  $g_1, g_2, r, \Delta_1$ , and  $\Delta_2$ , we chose “pixels” (pixel width lengths). The solution is approximate since the correspondence between  $(g_1, g_2)$  and the real-space coordinates  $(x_1, x_2, x_3)$  is not exact. Hereafter, the projection

**Table 1.** Reference Points Used in Figure 2 for Parameter Estimation.

Number	Longitude (°)	Latitude (°)	Elevation (m)	$g_1$ (pixels)	$g_2$ (pixels)
0	−3.702702	40.417105	670.96	—	—
1	−3.702299	40.417011	649.88	7,720	9,820
2	−3.70226	40.416784	649.48	14,130	7,390
3	−3.702243	40.416656	649.47	16,280	6,250
4	−3.702331	40.416656	649.36	17,900	6,760
5	−3.703183	40.416614	648.41	35,270	6,510
6	−3.703388	40.416607	648.32	37,420	5,740
7	−3.704202	40.416533	647.09	41,640	4,110
8	−3.704306	40.416523	646.90	42,000	3,910
9	−3.704986	40.416462	646.52	43,220	3,360
10	−3.705702	40.416366	647.19	43,780	3,030
11	−3.704675	40.416634	646.22	43,980	3,760
12	−3.704697	40.416781	645.75	45,290	3,870
13	−3.704874	40.416821	645.31	45,850	3,750
14	−3.704383	40.416941	646.43	46,550	4,040
15	−3.704121	40.417091	647.28	48,180	4,440
16	−3.70398	40.417147	647.50	48,990	4,720
17	−3.703635	40.417227	648.36	50,820	5,650
18	−3.701658	40.417159	651.70	2,490	4,430

using Equation (4) with the parameters of Equation (6) is called *basic* projection. In Robustness Test: Projection section, we show that small inaccuracies in parameter estimation do not significantly affect the results. For instance, the relative standard error increases from 7 percent to nine percent in the considered case.

*Crowd Coordinate Simulation*

Assessing the precision of the estimators requires either an annotated gigapixel image or a realistic simulation of the particle positions. Since the former was not available, we simulated  $N$  coordinates ( $g_1, g_2$ ) of a crowd, resembling the population pattern of Figure 1c. As the real crowd size is unknown, we chose to simulate  $N = 20,000$  points, which seems large enough for our purpose.

First, we outlined a window in the map area where the  $N$  points had to be simulated (red line in Figure 2a). The window coordinates were marked manually in Google Earth Pro following approximately the viewshed from the observer location. Note that the viewshed is not exact since there are

ledges of the building that do not appear in Google Earth Pro. Furthermore, we outlined some interior boundaries or “holes” in the window, where no people can be found, namely, two fountains, a metro station, and a statue.

The point pattern shown in Figure 3a was simulated using the R-spatstat package Version 1.37-0 (Baddeley, Rubak, and Turner 2015) as shown in the Online Appendix. Distinct points were not allowed to come closer than 0.5 m apart. The  $x_3$  coordinate of each of the  $N$  simulated points was determined using Google Elevation API. Each point in the simulated cloud was then projected using Equation (4), thus obtaining the point cloud shown in Figure 3c. The grid was superimposed using the approximations described below (Robustness Test: Projection section). More realistic crowd simulations are not necessary since our naive simulation adequately resembles the inhomogeneous population pattern in Figure 1c.

## Results

### *Empirical Assessment of the Precision of the Estimators through Monte Carlo Resampling*

Monte Carlo resampling on the simulated point cloud was used to assess the precision of  $\hat{N}$  and the performance of the  $\text{var}_{\text{Cav}}(\hat{N})$  estimator. A standard grid of quadrats (i.e., *CountEm* method, see Figure 3b) and a projected grid (i.e., *CountEm2* method as in Figure 3c) were applied with  $60^\circ$  tilt prior to sampling. The resampling procedure described by Cruz et al. (2015) was followed. Next, we recall the necessary notation:

- $Y = \{y_1, y_2, \dots, y_N\}$ : finite set representing a population of size  $N$  in a bounded area.
- $y_i \in Y$ :  $i$ th particle of the population.
- $J_0$ : fundamental tile or box of side length  $T$ .
- $z \in J_0$ : UR point in the fundamental tile.
- $\Lambda_z$ : UR systematic grid of quadrats, generated by shifting the lower-left corner of a quadrat from an arbitrary initial position in  $J_0$  into the UR point  $z$ , thus dragging the whole quadrat grid together. The quadrats have side length  $t$ .
- $Q = Q(Y \cap \Lambda_z)$ : sample size, namely, the total number of particles captured by the quadrats.

For each pair  $(t, T)$ , a total of  $K^2 = 32^2 = 1024$  replicated superimpositions of the grid  $\Lambda_z$  onto  $Y$  were generated, corresponding to  $K^2$  systematic

replications  $\{z_k, k = 1, 2, \dots, K^2\}$  of the point  $z$  within  $J_0$ . These  $K^2$  point locations were arranged in a random subgrid within  $J_0$  which should be expected to be more efficient than independent random locations as explained by Cruz et al. (2015). Thus,  $K^2 = 32^2 = 1,024$  simulations are sufficient to ensure that the empirical values are very close to their respective true values.

For each  $k$ , the corresponding sample total,

$$Q_k = Q(Y \cap \Lambda_{z_k}), \quad (7)$$

was computed automatically using the R-spatstat package (Baddeley et al. 2015). Hence, from Equation (1), we obtain

$$\hat{N}_k = (T/t)^2 \cdot Q_k. \quad (8)$$

The empirical mean and variance of  $\hat{N}$  were computed, respectively, as follows:

$$\mathbb{E}_e(\hat{N}) = K^{-2} \sum_{k=1}^{K^2} \hat{N}_k, \quad (9)$$

$$\text{Var}_e(\hat{N}) = K^{-2} \sum_{k=1}^{K^2} \left\{ \hat{N}_k - \mathbb{E}_e(\hat{N}) \right\}^2. \quad (10)$$

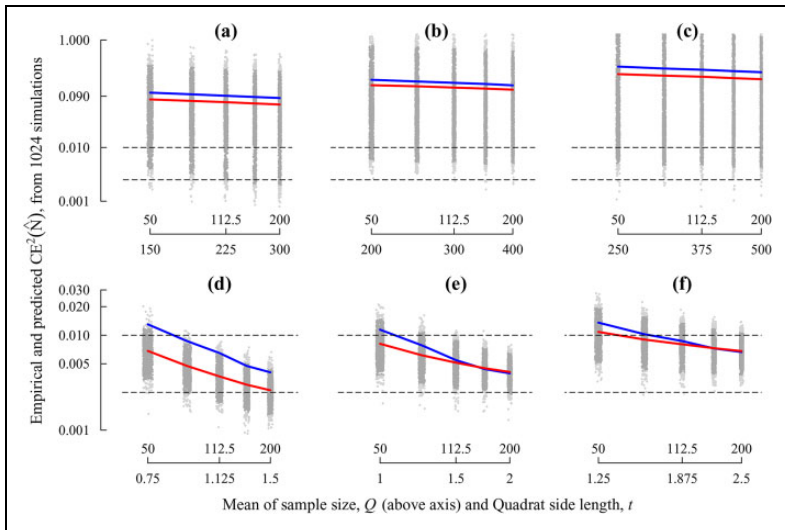
We also computed the corresponding  $K^2$  replicates of  $\{\text{var}_{\text{Cav}}(\hat{N}_k)\}$  using eq. 3 of Cruz et al. (2015). The empirical square coefficient of error:

$$\text{ce}_{\text{Cav}}^2(\hat{N}) = \frac{1}{N^2 K^2} \sum_{k=1}^{K^2} \text{var}_{\text{Cav}}(\hat{N}_k) \quad (11)$$

was compared with the corresponding empirical (“true”) value,

$$\text{CE}_e^2(\hat{N}) = \text{Var}_e(\hat{N})/N^2. \quad (12)$$

The results and the considered  $t$  and  $T$  values are shown in Figure 4. The grid units in the *CountEm* case are pixels. For *CountEm2*, sampling on the map as in Figure 3a and on the picture Figure 3c yield identical results assuming that the grid is transformed with the *exact* camera projection. For simplicity, we chose to sample on the map and hence, in that case,  $t$  and  $T$  are in meters. In the bottom row of Figure 4, the empirical error coefficient of the *CountEm2* method (blue line) is as expected in the 5–10 percent range, whereas the standard *CountEm* method (Figure 4a–c) presents error



**Figure 4.** (a–c) Monte Carlo results sampling with a standard grid of quadrats as in Figure 3b, for fundamental box side lengths  $T = 3,000$ ,  $4,000$ , and  $5,000$  pixels, respectively, and for different quadrat side lengths in each case. The equivalent mean sample sizes are also shown. The empirical square coefficient of error ( $=$  error variance divided by  $N^2$ ) is represented in blue, whereas the mean Cavalieri predictor is represented in red color. Gray dots represent all the replicated values of  $ce^2_{\text{Cav}}(\hat{N})$ . The dark gray dots lie between the 2.5 percent and 97.5 percent quantiles. The broken horizontal lines correspond to 5 percent and 10 percent coefficients of error, respectively. (d–f) Analogous data using a projected grid as in Figure 3c—here,  $T = 15$ ,  $20$ , and  $25$  m, respectively. Empirical coefficients of error are in the 30–40 percent range for *CountEm* and in the 5–10 percent for *CountEm2*. The Cavalieri estimator shows a reasonable performance in all the considered cases.

coefficients in the 30–40 percent range. Thus, the projected grid is able to correct the huge variance increase due to perspective artifacts. The variance estimator (red line) exhibits a reasonable performance in all the considered cases. These statements are based on the assumption that there is no human measurement error, thus, in practice, the errors could be slightly higher.

### Robustness Test: Projection

In the preceding section, we assumed that the grid could be transformed using the *exact* camera projection. For simplicity, sampling was performed

on the map Figure 3a instead of on the picture Figure 3c. However, in practice, some inaccuracies can arise in the calibration and/or grid projection processes. For instance, the reference points could be different, the elevation data could be inaccurate or unavailable, or the number of projected points per quadrat could be insufficient. Here, we examine the impact of such inaccuracies by comparing a *simplified* grid projection with the *exact* projection, using the simulated point cloud.

The following approximations are used in the *simplified* projection which we applied to the grid:

- We assumed that, for all the quadrats of the grid, only a common, average elevation was known. For the grid projection, the elevation,  $x_3$ , was set at 648 m in Equation (4).
- The following set of parameters was used in Equation (4):

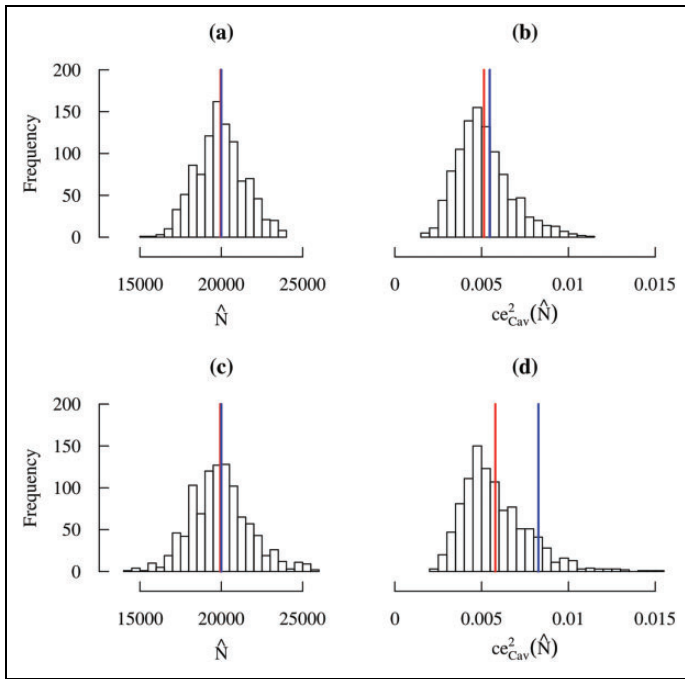
$$\begin{aligned} r &= 1,428.6 \text{ pixels} \\ \Delta_1 &= 2,584.6 \text{ pixels} \\ \Delta_2 &= 1,58.9 \text{ pixels} \end{aligned} \tag{13}$$

which were estimated using only six reference points, numbered 3, 6, 9, 12, 15, 18 in Table 1. They are plotted in cyan color in Figure 1a.

- We chose to project six points per quadrat, namely, the four vertices plus the two end points of the interrupted forbidden line segments. The quadrats were then approximated by connecting those six points with straight line segments. Note that only vertical straight lines are preserved in cylindrical projections.

On the other hand, the *basic* projection (defined in Cylindrical Projection: Parameter Estimation section) was applied to the simulated point cloud using the elevations given by Google Elevation API.

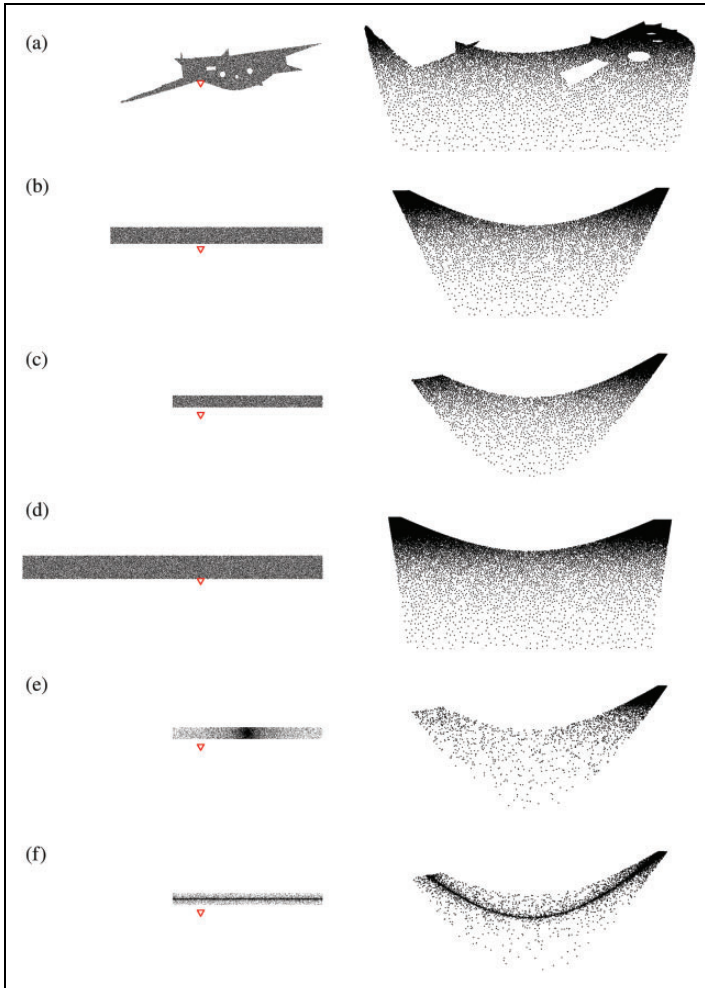
Figure 5 shows the results for  $\hat{N}$  and  $ce_{\text{Cav}}^2(\hat{N})$  with  $T = 20$  m and  $t = 1.5$  m, which are the parameters chosen in Population Sizing With the Gigapixel Picture section. The results using the *exact* projection (i.e., sampling on the map) are plotted in panels (a) and (b), and the ones for the *simplified* grid projection combined with *basic* point cloud projection in (c) and (d). As it can be seen, the *simplified* projection does not introduce appreciable bias. The empirical variance increases slightly, since the relative coefficient of error is 7 percent for the *exact* projection and 9 percent for the *simplified* projection. The  $ce_{\text{Cav}}^2(\hat{N})$  estimator performs only slightly worse.



**Figure 5.** Histograms of the Monte Carlo results for the 1,024 replications for  $T = 20$  m and  $t = 1.5$  m using exact (a and b) and simplified (c and d) projection, respectively. The mean of the 1,024  $\hat{N}$  Monte Carlo values is represented in (a) and (c) with a red line, whereas the blue line shows the “true” empirical value  $N = 20,000$ . Analogously, in (b) and (d), the red line stands for the mean Cavaliere predictor and the blue line for the empirical square coefficient of error.

### *Robustness Test: Population Size and Spatial Distribution*

The crowd point pattern simulation of Crowd Coordinate Simulation section had population size  $N = 20,000$  and a homogeneous, hard-core spatial distribution. Here, we analyze how the projection correction does help to reduce the error in other simulated point patterns, varying  $N$  and the spatial distribution. We considered the six simulations shown in Figure 6. They were simulated with the spatstat package (Baddeley et al. 2015) as described in the Online Appendix. The inhomogeneous distributions in Figure 6e–f are expected to be more inhomogeneous than real crowds, since no hard-core radius was considered.



**Figure 6.** Simulated point patterns as in Figure 3, with different crowd sizes and spatial distributions. The left column represents the particles on a map based on Figure 2a, and the right column the unwrapped cylindrical projection of the point patterns. (a)  $N = 2 \times 10^4$  points with hard-core distribution as in Figure 3. (b) Same as (a) but considering a rectangular area on the map. (c) Same as (b) but  $N = 10^4$  and halving the area of the rectangle. (d) Same as (b) but  $N = 4 \times 10^4$  and doubling the rectangular area. (e) Same as (c) but with inhomogeneous spatial distribution. The particle density decreases with the distance to the center of the rectangle. (f) Same as (e) but the particle density decreases with the distance to the central x-axis of the rectangle. The red triangle represents the position of the camera.



**Table 2.** Population Size Estimation Results with Projection Correction.

Image	$T$	$N$	$\widehat{N}_1$	$Q_1$	$ce_{Cav}(\widehat{N}_1)$	$CE_e(\widehat{N})$
(a)	150	20,000	20,800	208	.039	.062
(b)	150	20,000	19,500	195	.041	.045
(c)	$150/\sqrt{2}$	10,000	9,750	195	.047	.046
(d)	$150\sqrt{2}$	40,000	38,600	193	.046	.047
(e)	$150/\sqrt{2}$	10,060	9,750	195	.079	.109
(f)	$150/\sqrt{2}$	10,020	9,250	185	.147	.081
	Mean	18,347	17,942	195	.062	.063

**Table 3.** Population Size Estimation Results without Projection Correction.

Image	$T$	$N$	$\widehat{N}_1$	$Q_1$	$ce_{Cav}(\widehat{N}_1)$	$CE_e(\widehat{N})$
(a)	3,000	20,000	15,200	152	.147	.278
(b)	3,000	20,000	21,000	210	.305	.419
(c)	$3,000/\sqrt{2}$	10,000	6,450	129	.101	.395
(d)	$3,000\sqrt{2}$	40,000	36,400	182	.441	.811
(e)	$3,000/\sqrt{2}$	10,060	9,050	181	.405	.343
(f)	$3,000/\sqrt{2}$	10,020	4,150	83	.087	.785
	Mean	18,347	15,375	156	.431	.784

The empirical values  $CE_e(\widehat{N})$  corresponding to the six simulated “gigapixel” point patterns (Figure 6) are computed following the simulation procedure described in Empirical Assessment of the Precision of the Estimators Through Monte Carlo Resampling section. The results with projection correction (*CountEm2*) are shown in Table 2, whereas those obtained without projection correction (*CountEm*) are displayed in Table 3. Parameters  $t$  and  $T$  were chosen in order to obtain a theoretical mean sample size of  $Q = 200$ . Quadrat side length was set to  $t = 1.5$  m and  $t = 300$  pixels for *CountEm2* and *CountEm*, respectively. The values of separation between quadrat centers,  $T$ , are shown in the Tables. A singular crowd size estimation  $\widehat{N}_1$  with sample size  $Q_1$  and error estimation  $ce_{Cav}(\widehat{N}_1)$ , obtained with a single superimposition of the grid of quadrats, is also shown. The mean values are given in the last row of each table. Note that as usual,  $\text{Mean}\{CE(\widehat{N})\} = \sqrt{\text{Mean}\{\text{Var}(\widehat{N})\}}/N$ .

The six *CountEm2* empirical coefficients of error,  $CE_e(\widehat{N})$ , are below 11 percent showing the robustness of the method to variations in population size and spatial distribution. The corresponding *CountEm* values without projection correction are above 27 percent.

## Population Sizing with the Gigapixel Picture

The previous sections reveal that applying *CountEm2* with  $T = 20$  m and  $t = 1.5$  m using the *simplified* grid projection should provide an unbiased estimator of the crowd size in Figure 1c with empirical coefficient of error of around 9 percent. For simplicity, this was the chosen setup.

In this context, we consider a particle as the planar projection of a human head or a clearly distinguishable fragment of it. This is a reasonable choice since most of the bodies are occluded but almost all heads are visible. Furthermore, the forbidden line rule was used for unbiased manual counting in the sample (Cruz et al. 2015), that is, a human head was counted in a quadrat only if it had points in common with the quadrat but it did not hit the extended, red forbidden line of the quadrat.

Using a  $60^\circ$  tilt,  $Q = 166$  faces were counted in the 37 nonempty quadrats of Figure 7, yielding  $\hat{N} \approx 29,500$  and a predicted coefficient of error of 9 percent.

## Application to Real Crowds

Throughout the article, we have only estimated the number of particles clearly distinguishable in an image, ignoring those that have not been pictured. Therefore, in order to estimate the size of a real crowd, all the people in the crowd should be distinguishable in the image. This is technically feasible but can be expensive in the case that the crowd covers several streets, or if some people are occluded by trees or other objects. The gigapixel image considered in Figure 7 only covered a part of the demonstration, which continued along adjacent streets. The number of occluded people was neglected, although a few people might be hidden by the metro station and the statue. Moreover, only about 98 percent of the area of Figure 3a appears on the cylindrical projection images Figure 3b and c. Therefore, we would need additional images of the unobserved area and the adjacent streets in order to estimate the total size of the demonstration. In quantitative microscopy, the analogous problem is solved using systematically sampled images (figs. 2 and 4 in Cruz-Orive and Weibel 1981).

A comparison with official estimates given by police, media, and organizers is ill-posed because unbiased estimates obtained by proper sampling cannot be compared with “guesstimates.” In addition, *CountEm* directly estimates population size avoiding inaccurate area estimations. In fact, defining the area corresponding to a particle cloud is a nontrivial problem (Ripley and Rasson 1977). A rough estimation of the area observed in Figure 7 yields



**Figure 7.** Low-resolution version of the  $T = 20$  m and  $t = 1.5$  m sampling grid projected onto the gigapixel image (top). The quadrats marked with yellow arrowheads are magnified in the bottom row.

13,600 m<sup>2</sup>, which together with the result  $\hat{N} \approx 29,500$  leads to a density of about 2.2 people/m<sup>2</sup>. According to newspaper *EL PAIS*, the densities in this area were between 3 and 4 people/m<sup>2</sup>, which would lead to a crowd size between 40,800 and 54,400 people.

## Summary and Conclusion

Geometric sampling on still images using a square grid of square quadrats provides an unbiased population size estimator irrespective of population size and pattern. Only 50 – 100 manual counts are usually necessary to achieve relative standard errors of 5–10 percent. This can be done within a few minutes and it allows fast and reliable quantification by any users. The only practical limitation is the basic requirement that all the particles in the population should be unambiguously identifiable for counting. For large populations ( $N > 10^4$ ), as the one considered here, the requirement can be met using gigapixel pictures. However, the big perspective artifacts significantly increase the sampling variance. This can be circumvented by generating the sampling grid on a map and projecting it onto the gigapixel picture. We show that in our case, the relative standard error can be reduced from about 30 percent to near 7 percent. Calibration errors and projection inaccuracies show a low impact on the precision of the estimator. For  $T = 20$  m and  $t = 1.5$  m, the relative standard error increases only slightly, from 7 percent to 9 percent using the approximations considered in Robustness Test: Projection section. The robustness against variations in population size and spatial distribution has been shown in Robustness Test: Population Size and Spatial Distribution section. The coefficient of error remains below 11 percent for all the cases considered in Figure 6, using sample size  $Q \approx 200$ . Finally, we estimated the size of the large real crowd visible in Figure 1c, as  $\hat{N} \approx 29500$ , with an estimated relative standard error of only 9 percent.

## Declaration of Conflicting Interests


The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

## Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: The authors acknowledge financial support from the Spanish Project AYA2015-66357-R (MINECO/FEDER).

## ORCID iD

Marcos Cruz  <http://orcid.org/0000-0002-4767-530X>

Javier González-Villa  <http://orcid.org/0000-0001-8602-908X>

## Supplemental Material

Supplemental material for this article is available online.

## References

- Baddeley, A. J. 1999. "Spatial Sampling and Censoring." Pp. 37-78 in *Stochastic Geometry: Likelihood and Computation*, edited by O. E. Bandorff-Nielsen, W. S. Kendall, and M. N. M. van Lieshout. London, England: Chapman & Hall/CRC.
- Baddeley, A. J. and E. B. V. Jensen. 2005. *Stereology for Statisticians*. London, England: Chapman & Hall/CRC.
- Baddeley, A., E. Rubak, and R. Turner. 2015. *Spatial Point Patterns: Methodology and Applications with R*. Boca Raton, USA: CRC Press.
- Baddeley, A. and R. Turner. 2000. "Practical Maximum Pseudolikelihood for Spatial Point Patterns." *Australian & New Zealand Journal of Statistics* 42:283-322.
- Botta, F., H. S. Moat, and T. Preis. 2015. "Quantifying Crowd Size with Mobile Phone and Twitter Data." *Royal Society Open Science* 2: 150162. Retrieved August 9, 2018 (<http://dx.doi.org/10.1098/rsos.150162>).
- Cruz, M., D. Gómez, and L. M. Cruz-Orive. 2015. "Efficient and Unbiased Estimation of Population Size." *PLoS One* 10:1-14. Retrieved August 9, 2018 (<http://dx.doi.org/10.1371/journal.pone.0141868>).
- Cruz-Orive, L. M. 2017. "Stereology: A Historical Survey." *Image Analysis & Stereology* 36:153-77. Retrieved August 9, 2018 (<https://ias-iss.org/ojs/IAS/article/view/1767>).
- Cruz-Orive, L. M. and E. R. Weibel. 1981. "Sampling Designs for Stereology." *Journal of Microscopy* 122:235-57.
- Descamps, S., A. Béchet, X. Descombes, A. Arnaud, and J. Zerubia. 2011. "An Automatic Counter for Aerial Images of Aggregations of Large Birds." *Bird Study* 58:302-8.
- EL PAIS. Recuento de manifestantes en la 'marcha del cambio'. Retrieved August 9, 2018 ([https://elpais.com/elpais/2015/01/31/media/1422734596\\_211375.html](https://elpais.com/elpais/2015/01/31/media/1422734596_211375.html)).
- Gundersen, H. J. G. 1977. "Notes on the Estimation of the Numerical Density of Arbitrary Profiles: The Edge Effect." *Journal of Microscopy* 111:219-23.
- Gundersen, H. J. G., E. B. V. Jensen, K. Kiêu, and J. Nielsen. 1999. "The Efficiency of Systematic Sampling in Stereology—Reconsidered." *Journal of Microscopy* 193:199-211. Retrieved August 9, 2018 (<http://dx.doi.org/10.1046/j.1365-2818.1999.00457.x>).
- Howard, C. V. and M. G. Reed. 2005. *Unbiased Stereology. Three-dimensional Measurement in Microscopy*. 2nd ed. Oxford, England: Bios/Taylor & Francis.

- Idrees, H., I. Saleemi, C. Seibert, and M. Shah. 2013. "Multi-source Multi-scale Counting in Extremely Dense Crowd Images." Pp. 2547-54 in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, edited by Eric Mortensen and Sanja Fidler. Washington, DC: IEEE.
- Jacobs, H. 1967. "To Count a Crowd." *Columbia Journalism Review* 6:37.
- Lempitsky, V. and A. Zisserman. 2010. "Learning to Count Objects in Images." Pp. 1324-32 in *Advances in Neural Information Processing Systems*, edited by J.D. Lafferty, C.K.I Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta Belfast, UK: Curran Associates, Inc.
- Lynch, T. P., R. Alderman, and A. J. Hobday. 2015. "A High-resolution Panorama Camera System for Monitoring Colony-wide Seabird Nesting Behaviour." *Methods in Ecology and Evolution* 6:491-99.
- Morán, A. Gigapan podemos puerta del sol. 2015. Retrieved ([http://lab.elespanol.com/estaticos/gigapan\\_sol](http://lab.elespanol.com/estaticos/gigapan_sol)).
- Rodriguez, M., I. Laptev, J. Sivic, and J. Y. Audibert. 2011. "Density-aware Person Detection and Tracking in Crowds." Pp. 2423-30 in *2011 International Conference on Computer Vision*, edited by Dimitris N. Metaxas, Long Quan, Alberto Sanfeliu, Luc J. Van Gool. Washington, DC: IEEE.
- Ripley, B. D. and J.-P. Rassin. 1977. "Finding the Edge of a Poisson Forest." *Journal of Applied Probability* 14:483-91.
- Watson, R. and P. Yip. 2011. "How Many Were There When It Mattered?" *Significance* 8:104-7.
- Zhang, C., H. Li, X. Wang, and X. Yang. 2015. "Cross-scene Crowd Counting via Deep Convolutional Neural Networks." Pp. 833-41 in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, edited by Mortensen Eric, Fidler Sanja. Washington, DC: IEEE.
- Zhang, S., R. Benenson, M. Omran, J. Hosang, and B. Schiele. 2016. "How Far Are We from Solving Pedestrian Detection?" Pp. 1259-67 in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, edited by Mortensen Eric, Saenko Kate. Washington, DC: IEEE.

## Author Biographies

**Marcos Cruz** is an associate professor in Statistics at the University of Cantabria. His research interests are Image analysis, statistical methods, stereology, sampling, and their applications to several topics such as cosmology, ecology or sociology.

**Javier González Villa** received his BE degree in Computer Engineering in 2013 and his MSc in Mathematics and Computing in 2014 from the University of Cantabria (UC), Spain. He is a PhD student in image analysis and stereology since 2014.