Surveying and benchmarking techniques to analyse DNA gel fingerprint images

Jónathan Heras, César Domínguez, Eloy Mata and Vico Pascual

Abstract

DNA fingerprinting is a genetic typing technique that allows the analysis of the genomic relatedness between samples, and the comparison of DNA patterns. The analysis of DNA gel fingerprint images usually consists of five consecutive steps: image pre-processing, lane segmentation, band detection, normalization and fingerprint comparison. In this article, we firstly survey the main methods that have been applied in the literature in each of these stages. Secondly, we focus on lane-segmentation and band-detection algorithms—as they are the steps that usually require user-intervention—and detect the seven core algorithms used for both tasks. Subsequently, we present a benchmark that includes a data set of images, the gold standards associated with those images and the tools to measure the performance of lane-segmentation and band-detection algorithms. Finally, we implement the core algorithms used both for lane segmentation and band detection, and evaluate their performance using our benchmark. As a conclusion of that study, we obtain that the average profile algorithm is the best starting point for lane segmentation and band detection.

Key words: DNA fingerprinting; benchmarking; lane segmentation; band detection; survey

Introduction

DNA fingerprinting is a technique for comparing DNA patterns that allow the analysis of the genomic relatedness among different samples, as well as to type and classify them. There are multiple DNA fingerprinting techniques, and the choice of which of them we must use depends on their applications (medical diagnosis, forensic science, parentage testing, food industry, agriculture and many others) [1].

After capturing DNA gel fingerprint images (also known as gel images), the process to analyse them can be split into five steps (Figure 1). First, the image is pre-processed to remove noise and fix distortions. Subsequently, the lanes of the image are segmented, and for each of those lanes (lane images) the bands are detected. Because the band positions of a lane are influenced by experimental conditions, a normalization step is carried out to compare banding patterns within the same gel and also from different gels. Finally, the similarity among lanes (based on banding patterns) is computed and graphically represented by means of a dendrogram (a hierarchical tree).

Several software tools implement the workflow to analyse gel images [2]; hence, the processing of those images is highly automated. However, there are two steps that usually require user intervention: lane segmentation and band detection. In the former, the user might need to manually add lanes, remove some of the automatically segmented lanes or adjust the thickness and curvature of the detected lanes. In the latter, the user might need to add or remove bands. These two manual corrections are tedious and time consuming, and, therefore, algorithms that reduce this effort are desirable.

In the literature, several approaches have been studied to tackle lane segmentation and band detection; however, a
comparison of the different methods does not exist. The reason is 2-fold: the lack of a common benchmark to measure the performance of those methods, and the unavailability of the implementation of most algorithms. In this article, we deal with these two problems. The main contributions of this work are:

1. A survey of the fundamental techniques used in each stage of DNA gel fingerprint analysis (see ‘Survey of methods’ section)—special attention is paid to lane-segmentation and band-detection methods.
2. A benchmark for analysing gel and lane images, including a data set of images, the gold standards associated with those images and a set of tools that has been designed to obtain different quantitative measures from the data set (see ‘Benchmarking: data set, gold standard, and analysis tools’ section).
3. The implementation and comparison of the core algorithms used in the segmentation of lanes and the detection of bands (see ‘Implementation and evaluation of lane-segmentation and band-detection algorithms’ section).

**Survey of methods**

In this section, we provide a survey of the fundamental techniques used in the five stages of DNA gel fingerprint analysis. We screened PubMed Central and Google Scholar looking for corpora publications, and used the Google search engine to create a list of papers devoted to analyse gel images—the search strategy that we have followed is described in Supplementary Appendix A. This search produced 35 papers explaining different approaches.

As we have previously explained, we are mainly focused on the lane-segmentation and band-detection stages—because they are the two most time-consuming and tedious tasks. However, and for the sake of completeness, we also survey the main techniques used in the other steps of the procedure.

**Image pre-processing**

DNA gel fingerprint images might suffer from various types of distortions, including geometric distortion of the whole image, horizontal lane deformation (smiling), salt-and-pepper noise, non-uniform background or low/high contrast bands. Therefore, most approaches perform some safe image transformations before the actual analysis of gel images. Namely, the most usual techniques are filtering, background subtraction, smiling correction, morphological closing and deconvolution.

**Filtering**

The application of different filters to an image might smooth it, enhance the edges in the image and reduce its noise. Several filters have been used in the literature of gel-images analysis: the average filter [3–7] reduces the noise of the images, the notch filter is applied to smooth gel images [8, 9], the minimum filter reduces high-frequency noise [5], the Gaussian filter enhances the edges of the images [10, 11], the least-square filter reduces noise and smooths the images [12], the low pass homomorphic filter enhances the images [13], and the match filter also enhances the images [14].

**Background subtraction**

This technique removes local background differences. Several papers, see [12, 13], apply the rolling ball mechanism [15] to...
remove the background from a gel image. In \[16\], the background fog is removed applying a maximum filter and, subsequently, a minimum filter. An automatic threshold is applied to equalize the grey values of the background in \[14\]. A polynomial function is used to model the background of images in \[17\] for latter subtraction, and a ‘Top-Hat Transform’ is used in \[18\] to subtract the background.

Smiling correction
This mechanism is used to correct gel distortion and smiling effects. In \[12\], they use a bounding box with distortion nodes to border the relevant part of the gel and to correct gel distortion and smiling effects. The method presented in \[16\] to correct the smiling effect consists in detecting and straightening a pair of bands common to most of the lanes. In \[9\], the smiling effect is fixed using a grid that captures the shape of distortions.

Morphological closing
This technique is used to remove noise. Different structuring elements can be used for morphological closing: a circular structure element of 5 pixel radius is used in \[19\], a square structure element is used in \[20\], a rectangular structure element is used in \[6, 11\] and a one-dimensional structural element parallel to the lanes is applied in \[4\].

Deconvolution
This method sharpens images and enhances the contrast of bands; however, this technique also increases the noise of the image. This mechanism has been used in \[11, 12\].

Lane segmentation and band detection
After a gel image has been pre-processed, the lanes of such an image are segmented, and afterwards, the bands of the segmented lanes are detected. The same intuitive idea is applied both in the segmentation of lanes and in the detection of bands. Because lane areas are covered with biological material, they appear lighter than the empty background areas between lanes; hence, strong intensity transitions between lanes and background are expected when moving horizontally across the image—analogously for bands when moving vertically across a lane. This idea is captured using a vertical (or horizontal in the case of bands) projection profile, and subsequently, obtaining the local peaks of such a profile (Figure 2).

In the literature, several projection profiles have been studied—we only consider here the definition of vertically projected profiles, and the definition of horizontally projected profiles is analogous. Given an image \(I\) with \(N\) columns and \(M\) rows of pixels, the vertical projection profile of \(I\) is an array of \(N\) elements that can be constructed using different methods—we will use \(I_{ij}\) to denote the intensity of the pixel located at column \(i\) and row \(j\) of \(I\).

**Average profile** \[10, 12–14, 16, 20–27\] The i-th element of the average profile—denoted by \(P_{\text{AVG}}(i)\)—is computed using the formula:

\[
P_{\text{AVG}}(i) = \frac{1}{M} \sum_{j=1}^{M} I_{ij}
\]

**Derivative profile** \[3, 7, 9, 17, 28–30\] The i-th element of the derivative profile—denoted by \(P_{\text{DER}}(i)\)—is computed using the formula:

\[
P_{\text{DER}}(i) = \sum_{j=1}^{M} I_{i+1,j} - I_{ij}
\]

**Sum profile** \[4, 5, 8, 19, 28, 31, 32\] The i-th element of the sum profile—denoted by \(P_{\text{SUM}}(i)\)—is computed using the formula:

\[
P_{\text{SUM}}(i) = \sum_{j=1}^{M} I_{ij}
\]

**Binary** \[6, 11, 18, 19, 33, 34\] The i-th element of the binary profile—denoted by \(P_{\text{BIN}}(i)\)—is computed after binarizing the image applying a threshold, and subsequently, using the formula:

\[
P_{\text{BIN}}(i) = \sum_{j=1}^{M} \chi_{ij}
\]

where \(\chi_{ij}\) is the value of the pixel located at column \(i\) and row \(j\) of the binarized version of \(I\). The computation of the threshold can be carried out using different methods like Otsu \[11\] or Kapur \[6\].

**Maximum profile** \[35, 36\] The i-th element of the maximum profile—denoted by \(P_{\text{MAX}}(i)\)—is computed using the formula:

\[
P_{\text{MAX}}(i) = \max_{j=1}^{M} I_{ij}
\]

Figure 2. Average profile from a lane image. The horizontal lanes indicate the bands located from the peaks, the dotted square is a local peak coming from noise and the non-dotted square is a peak that comes from an uncertain band.
The height threshold is usually left to the user. This value varies from image to image. Therefore, the task of fixing the location of a band. However, this criterion has two disadvantages: it can exclude low-intensity bands (see Supplementary Appendix B). In the surveyed papers are summarized in Figure 2, some of the local peaks come from noise, and they are excluded by using a minimum height criterion: the value of the local peak must be higher than a fixed minimum to be considered as the location of a band. However, this criterion introduces the introduction of a height threshold [12, 16, 35]. As can be seen in Figure 2, some of the local peaks come from noise, and they are excluded by using a minimum height criterion: the value of the local peak must be higher than a fixed minimum to be considered as the location of a band. However, this criterion has two disadvantages: it can exclude low-intensity bands (see the non-dotted square in Figure 2), and the optimum height value varies from image to image. Therefore, the task of fixing the height threshold is usually left to the user.

Normalization

Owing to the fact that the band positions of a lane are influenced by experimental conditions, a normalization step is required to refine the lane-segmentation and band-detection tasks. The enhancements introduced in the surveyed papers are summarized in Supplementary Appendix B. A common improvement in the band-detection algorithms is the introduction of a height threshold [12, 16, 35]. As can be seen in Figure 2, some of the local peaks come from noise, and they are excluded by using a minimum height criterion: the value of the local peak must be higher than a fixed minimum to be considered as the location of a band. However, this criterion has two disadvantages: it can exclude low-intensity bands (see the non-dotted square in Figure 2), and the optimum height value varies from image to image. Therefore, the task of fixing the height threshold is usually left to the user.

**Standard derivation profile** [37] The i-th element of the standard derivation profile—denoted by $P_{\text{STD}}(i)$—is computed using the formula:

$$P_{\text{STD}}(i) = \left( \frac{1}{N-1} \sum_{j=1}^{M} (I_{ij} - \bar{I})^2 \right)^{1/2}$$

where $I_{ij} = \frac{1}{N} \sum_{j=1}^{M} I_{ij}$.

**Derivative of standard derivation profile** [37] The i-th element of the standard derivation derivative profile—denoted by $P_{\text{STDDER}}(i)$—is computed using the formula:

$$P_{\text{STDDER}}(i) = P_{\text{STD}}(i + 1) - P_{\text{STD}}(i)$$

These seven profiles are the core for the construction of more complex algorithms that refine the lane-segmentation and band-detection tasks. The enhancements introduced in the surveyed papers are summarized in Supplementary Appendix B.

A common improvement in the band-detection algorithms is the introduction of a height threshold [12, 16, 35]. As can be seen in Figure 2, some of the local peaks come from noise, and they are excluded by using a minimum height criterion: the value of the local peak must be higher than a fixed minimum to be considered as the location of a band. However, this criterion has two disadvantages: it can exclude low-intensity bands (see the non-dotted square in Figure 2), and the optimum height value varies from image to image. Therefore, the task of fixing the height threshold is usually left to the user.

**Fingerprint comparison**

The process to compare fingerprints (i.e. lanes) consists of two steps: the computation of similarity matrices, and the construction of dendrograms.

**Similarity matrices**

Given a list of $n$ lanes, $L$, the similarity matrix of $L$ is an $n \times n$ matrix where the element of row $i$ and column $j$ encodes the similarity between the i-th and j-th lanes of $L$. There are two approaches to calculate the similarity between lanes: band based and curve based [12]—a search in PubMed shows that both approaches are equally used in the literature (Supplementary Appendix A). In the former approach, the similarity between two lanes is calculated as a coefficient based on the number of matching and non-matching bands. In the latter approach, the similarity is determined using a correlation coefficient computed from the projection profiles (also known as densitometric curves) of the lanes.

**Band-based methods.** The comparison of lanes using band-based methods is a two-step mechanism: (i) matching is performed between the bands of two lanes, and (ii) the similarity of two lanes is computed based on the number of matching and/or non-matching bands.

In the first step, a tolerance value is introduced. This value indicates the maximum distance allowed between two bands to be considered as matching. Under this criterion, two (or more) bands on one lane might be eligible for matching with the same band on another lane (Figure 3). Two alternatives are considered to solve this problem: closest band matching or first band matching. In the former, the two bands that have the shortest distance are matched; in the latter, the first candidate that is encountered is matched (Figure 3).

Once the bands of two lanes are matched, the similarity between them can be computed using different coefficients. The most common band-based coefficients are provided in Table 1.

**Curve-based methods.** The curve-based coefficients work with the densitometric curves associated with the different lanes—as we have seen in ‘Lane segmentation and band detection’ section,
different densitometric curves can be associated with a given lane. The most common curve-based coefficients used in the literature are summarized in Table 2.

Band-based versus curve-based methods. These two kinds of methods have their pros and cons. The advantage of curve-based coefficients is that they are less subjective than the band-based coefficients: band detection and tolerance fixation (two steps that require user intervention) are not required in curve-based methods, but they are necessary for band-based coefficients. However, the curve-based coefficients never show perfect matches—perfect matches are possible using the band-based coefficients. The advantage of band-based coefficients is that they provide a better control of the results (the bands selected from a lane can be manually modified by the user, but the densitometric curve cannot be altered).

Band- and curve-based methods are the traditional approach to compare fingerprints. Recently, a new approach has been proposed to classify 2D-gel images based on image texture [45]. Such a method could be extrapolated to classify fingerprints. The texture-based method can be seen as an improvement of the curve-based approach because it describes images using not only the information from the densitometric curve but also other features. Then, as in the case of the curve-based method, the texture-based approach will be less subjective than the band-based approach, but it will not show perfect matches. It remains as further work to compare the results that can be obtained with the three methods (band, curve and texture based).

Table 1. Band-based methods, their associated formula and the percentage of papers that use them

<table>
<thead>
<tr>
<th>Method</th>
<th>Formula</th>
<th>Papers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dice [40]</td>
<td>$\frac{2 \cdot b_i}{b_i + b_j}$</td>
<td>72%</td>
</tr>
<tr>
<td>Jaccard [41]</td>
<td>$\frac{b_i}{b_i + b_j - b_i b_j}$</td>
<td>12%</td>
</tr>
<tr>
<td>Different bands [12]</td>
<td>$1 - (b_i + b_j - 2b_i)$</td>
<td>8%</td>
</tr>
<tr>
<td>Ochiai [42]</td>
<td>$\frac{b_i}{\sqrt{b_i b_j}}$</td>
<td>7%</td>
</tr>
</tbody>
</table>

Note. The following notation is used: given two lanes $L_i$ and $L_j$, $b_i$ and $b_j$ is the number of common bands (i.e. matched bands) that appear in the lanes $L_i$ and $L_j$, $b_i$ is the number of bands that appear in $L_i$ and $b_j$ is the number of bands that appear in $L_j$.

Table 2. Curve-based methods, their associated formula and the percentage of papers that use them

<table>
<thead>
<tr>
<th>Method</th>
<th>Formula</th>
<th>Papers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pearson coefficient</td>
<td>$\frac{\sum_{i=1}^{n} x_i y_i - n \cdot \overline{x} \overline{y}}{\sqrt{\sum_{i=1}^{n} x_i^2 - n \cdot \overline{x}^2} \sqrt{\sum_{i=1}^{n} y_i^2 - n \cdot \overline{y}^2}}$</td>
<td>72%</td>
</tr>
<tr>
<td>Euclidean distance</td>
<td>$\sqrt{\sum_{i=1}^{n} (x_i - \overline{X})^2}$</td>
<td>18%</td>
</tr>
<tr>
<td>Cosine correlation</td>
<td>$\frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \sqrt{\sum_{i=1}^{n} y_i^2}}$</td>
<td>7%</td>
</tr>
</tbody>
</table>

Note. The following notation is used: given two lanes $L_i$ and $L_j$, with height $n$, their densitometric curves are two arrays of $n$ values where $x_i$ and $y_i$ are the ith values of the densitometric curves of $L_i$ and $L_j$, respectively.

Clustering algorithms

The similarity matrices are fed as input to hierarchical clustering algorithms [46]. These algorithms are used to visualize the relations among fingerprints using a dendrogram—a kind of hierarchical tree. The construction of dendrograms follows an iterative process: at each step, the nearest two clusters (sets of fingerprints) are combined into a higher-level cluster. The difference among the methods relies on how the distance between the new clusters is recomputed. The main methods used in the literature are summarized in Table 3. Alternatively to hierarchical clustering, some papers apply the neighbour joining algorithm to construct phylogenetic trees [47].

Benchmarking: data set, gold standard, and analysis tools

The workflow to analyse gel images could be fully automated—just picking an algorithm for each stage—however, the results obtained automatically are unlikely to be precise. In particular, to obtain accurate results, it is usually necessary to carry out some adjustments in the lane-segmentation and band-detection steps. Therefore, it is relevant to know what are the lane-segmentation and band-detection algorithms that reduce the user interaction.

In general, the evaluation of segmentation/detection algorithms requires three ingredients: a data set of images (e.g. the face recognition data set [48], Berkeley segmentation data set for natural images [49], the data set of macrobiological structures [50] or the UCSB biosegmentation benchmark [51]), a gold standard that fixes the optimum segmentation (the gold standard is manually provided by experts) and a set of metrics to compare the results obtained with the segmentation/detection algorithms and the gold standards.

In this section, we present a benchmark to test algorithms for the segmentation of lanes in gel images, and the detection of bands in lane images. The benchmark can be downloaded from http://www.unicrija.es/cu/joheras/surveying/. The instrumental tool that we have used to create the benchmark is ImageJ [52]: a freely available Java platform for image processing that has been widely used in several contexts, and that can be easily extended by means of plug-ins. Using ImageJ, we have defined the gold standards of the data set. In addition, we have extended ImageJ with two plug-ins to measure the

Table 3. Linkage methods, their associated formula and the percentage of papers that use them

<table>
<thead>
<tr>
<th>Method</th>
<th>Formula</th>
<th>Papers (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>UPGMA</td>
<td>$d(X, Y) = \frac{1}{m \cdot n} \sum_{i=1}^{m} \sum_{j=1}^{n} d(x_i, y_j)$</td>
<td>27</td>
</tr>
<tr>
<td>Single (minimum) linkage</td>
<td>$d(X, Y) = \min(d(x, y))$ where $x \in X, y \in Y$</td>
<td>24</td>
</tr>
<tr>
<td>Complete (maximum) linkage</td>
<td>$d(X, Y) = \max(d(x, y))$ where $x \in X, y \in Y$</td>
<td>18</td>
</tr>
<tr>
<td>Ward</td>
<td>$d(X, Y) = \frac{n_{xy}}{n_x n_y}</td>
<td>m_x - m_y</td>
</tr>
</tbody>
</table>

Note. The following notation is used: $X$ and $Y$ are clusters, $d(X, Y)$ is the similarity between the two clusters, $d(x, y)$ is the similarity between two objects of different clusters, $n_{xy}$ is the number of elements of the cluster $X$ and $m_x$ is the centre of the cluster $X$. 
performance, using different metrics, of lane-segmentation and band-detection algorithms.

Data set

The data set consists of 50 gel images and 121 lane images saved using the tiff format. The data set of gel images consists of 24 images of good quality, 17 of intermediate quality and 9 of bad quality—the quality of the gel images was based on the straightness of lanes, and the contrast and noise of the images. In the case of lane images, the data set contains 42 good images, 66 intermediate images and 13 bad images—the quality of the lane images is based on the opinion of experts, the migration of the lane and the contrast of the bands.

The images of the data set were taken from agarose PFGE gels. These gels were prepared with agarosa D-5 (Pronadisa, Conda) in 0.5X Tris-Boric-EDTA (TBE). In each gel at least two lanes were placed with molecular weight markers (Lambda Ladder PFGE Marker, Middle Range PFGE Marker or Low Range PFGE Marker) and the plugs of the test samples were placed in the remaining lanes. Electrophoresis was carried out in CHEF-DR II Drive Module (BioRad) machine with 2 L of 0.5X TBE with a spatula tip of thiourea. The conditions were different according to the enzyme and bacteria used. Gels were stained with an aqueous ethidium bromide solution (10 ml of ethidium bromide in 200 ml of distilled water) by immersion for 20 min under stirring. Gels were visualized with ultraviolet light and were photographed with Image Store 5000 UVP, thanks to the software ChemiGenius (GenSnap from SynGene).

Gold standards

The gold standard of the data set has been created using the ROI Manager tool (Figure 4) of ImageJ. The ROI Manager allows the user to fix a set of regions of interest (ROIs)—such regions might have different shapes (e.g. rectangle, oval, polygon or free shape)—and save it as a zip file for later use. Using this tool, four biological experts in the processing of PFGE images have manually segmented, by consensus, the gel and lane images of the data set, as a result, a set of gold standards have been created. In particular, 677 lanes and 1818 bands have been manually segmented respectively for the gel and lane images.

Analysis tools

We have extended ImageJ with two plug-ins that allow the evaluation of lane-segmentation and band-detection algorithms. These plug-ins are called LaneSegPerformanceJ and BandSegPerformanceJ. 

LaneSegPerformanceJ

This plug-in serves to measure the performance of lane-segmentation algorithms in terms of two different criteria: (i) the number of lanes that must be added or removed from the segmentation, and (ii) the adjustment that is necessary for the segmented lanes.

To measure the performance of segmentation algorithms regarding the former criteria, the input of LaneSegPerformanceJ is 3-fold: a gel image, the associated gold standard and the segmentation of the gel. The output generated by LaneSegPerformanceJ is the set of measures provided in the centre column of Table 4 (for more information about these measures, please refer to [53, 54]). The measures presented in such a table are based on the following values.

• True Positive (TP): the number of lanes of the gold standard that are located by the segmentation.
• False Positive (FP): the number of segmented lanes that do not correspond to any lane of the gold standard, and those segmented lanes that correspond to lanes of the gold standard that have been previously detected by other lanes of the segmentation.
• False Negative (FN): the number of lanes of the gold standard that are not located by the segmentation.
• True Negative (TN): the number of regions of the gel that do not contain either lanes of the gold standard or segmented lanes.

Unfortunately, the above values and the metrics presented in Table 4 are not intuitive for most experimental scientists, and an alternative formulation is preferred by many investigators, as shown in a series of recent publications [55–62]. In the alternative formulation, \( N^+_N \) is the total number of lanes of the gold standard, \( N^+_N \) is the number of true lanes of the gold standard predicted to be lanes by the segmentation (i.e. the TP value), \( N^-_N \) is the number of true lanes of the gold standard that are not located by the segmentation (i.e. the FN value), \( N^+\_N \) is the number of segmented lanes that do not belong to the gold standard (i.e. the FP value) and \( N^-\_N \) is the number of regions of the gel that do not contain either lanes of the gold standard or segmented lanes (i.e. the TN value). Using this alternative formulation, the output generated by LaneSegPerformanceJ is given by the set of measures provided in the right column of Table 4. A crystal clear interpretation of the different metrics is provided in [55–62].

For the latter criteria, i.e. to measure the adjustment that is necessary for the segmented lanes, LaneSegPerformanceJ takes the same input and produces the same output as explained in the previous case. However, the measures presented in Table 4 are based on the area correctly segmented for each lane.

• True Positive (TP or \( N^+_N \)): the number of pixels that belong both to the gold standard and the segmentation.
• False Positive (FP or \( N^-\_N \)): the number of pixels that belong to the segmentation but not to the gold standard.
• False Negative (FN or \( N^-\_N \)): the number of pixels that belong to the gold standard but not to the segmentation.
• True Negative (TN or \( N^-\_N \)): the number of pixels that neither belong to the gold standard nor to the segmentation.

BandSegPerformanceJ

The criteria used by this plug-in to measure the performance of band-detection algorithms is based on the number of bands that must be manually added or removed after the automatic detection.

The input of BandSegPerformanceJ is 3-fold: a lane image, the associated gold standard and the position of the detected bands in the lane (provided by a set of points). The output generated by BandSegPerformanceJ is the same set of measures generated in the LaneSegPerformanceJ plug-in, but computed using the following values.

• True Positive (TP or \( N^+_N \)): the number of bands of the gold standard that are located by the estimation.
• False Positive (FP or \( N^-\_N \)): the number of points of the estimation that do not correspond to any band of the gold standard, and those estimated points that correspond to bands that have been previously detected by other point of the estimation.
• False Negative (FN or \( N^-\_N \)): the number of bands of the gold standard that are not located by the estimation.
• True Negative (TN or N:\): the number of regions of the lane that do not contain either bands or estimated points.

As we have explained in 'Lane segmentation and band detection' section, different height threshold can be fixed to detect bands; hence, it might be interesting to measure the evolution of band-detection algorithms when altering such a threshold. BandSegPerformanceJ can be used to achieve this goal. To this aim, the plug-in takes as input a lane image, the associated gold standard and a batch of successive band detections. The output produced by the plug-in is the receiver operating characteristic (ROC) curve [63] (and the area under the ROC curve, also known as AUROC) associated with the batch, and the above mention measures for each individual detection of the batch.

Implementation and evaluation of lane-segmentation and band-detection algorithms

In this section, we present the implementation of the core algorithms for lane segmentation and band detection presented in 'Lane segmentation and band detection' section. Additionally, we evaluate them using the benchmark introduced in 'Benchmarking: data set, gold standard, and analysis tools' section.

Implementation

We have implemented two ImageJ plug-ins called LaneManagerJ and BandManagerJ—they can be downloaded from http://www.unirioja.es/cu/woheras/surveying/.
The LaneManagerJ plug-in takes as input a gel image, and allows the user to segment the lanes of such an image choosing one of the seven algorithms based on the seven profiles presented in ‘Lane segmentation and band detection’ section (average, binary, derivative, maximum, STD, STD-derivative and sum). The output produced by this plug-in is a set of regions (rois) that can be stored as a zip file—this zip file can be fed as input to the analysis tools presented in 'Benchmarking: data set, gold standard, and analysis tools' section. Likewise, the gel image, the gold standard and the lane segmentation are provided as input to LaneSegPerformanceJ that evaluates how good is the segmentation. The workflow for band detection is analogous using BandManagerJ instead of LaneManagerJ, and BandSegPerformanceJ instead of LaneSegPerformanceJ.

We include a statistical evaluation in the study. In the literature, a parametric test is preferred for model comparison when the necessary requirements are satisfied, but a non-parametric test is also acceptable when the distribution does not fulfil the assumptions [64-66]. Therefore, we firstly use analysis of variance with repeated measures to test whether there are differences on the evaluated methods. Secondly, we compare each pair of methods with a paired t-test using Bonferroni correction. When parametric conditions are not verified, we take into account the corresponding non-parametric tests (i.e. Friedman or Wilcoxon tests). These last comparisons allow us to sort (from best to worst) the methods according to a studied characteristic. A symbol > between two methods will mean that a significant difference between two consecutive methods has been found.

### Evaluation of lane-segmentation algorithms

As we have explained in ‘Benchmarking: data set, gold standard, and analysis tools’ section, the performance of lane-segmentation algorithms is evaluated regarding two criteria: the correct detection and the correct segmentation of lanes—these two criteria measure respectively how many lanes must be added or removed, and the adjustment of the thickness and shape of the lanes. We start by analysing the seven algorithms implemented in LaneManagerJ with respect to the lane-detection criterion.

Table 5 includes the means and standard deviations of the different algorithms to detect lane positions. There exist significant differences among the different methods in all the studied aspects. The average algorithm is the method with significant less FPs—i.e. the method that requires less user-intervention.

![Workflow to evaluate the performance of a lane-segmentation algorithm using our benchmark and LaneManagerJ. A gel image is given as input to the ROIManager tool of ImageJ where an expert defines the gold standard. In addition, the same image is the input of LaneManagerJ where the user can segment the lanes of the gel image using the available methods. Subsequently, the gel image, the gold standard and the lane segmentation are provided as input to LaneSegPerformanceJ that evaluates how good is the segmentation. The workflow for band detection is analogous using BandManagerJ instead of LaneManagerJ, and BandSegPerformanceJ instead of LaneSegPerformanceJ.]

![Diagram](https://example.com/diagram.png)

Table 4. Measures generated by the LaneSegPerformanceJ and BandSegPerformanceJ plug-ins

<table>
<thead>
<tr>
<th>Measure</th>
<th>Value using traditional formulation</th>
<th>Value using alternative formulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>TP + FN</td>
<td>N'</td>
</tr>
<tr>
<td>Negative</td>
<td>FP + TN</td>
<td>N'</td>
</tr>
<tr>
<td>Accuracy</td>
<td>( \frac{TP + TN}{TP + FN + FP + TN} )</td>
<td>( 1 - \frac{N' + N''}{N' + N''} )</td>
</tr>
<tr>
<td>Precision</td>
<td>( \frac{TP}{TP + FN} )</td>
<td>( N' )</td>
</tr>
<tr>
<td>Sensitivity/reCALL</td>
<td>( \frac{TP}{TP + FN} )</td>
<td>( \frac{N' + N''}{N'} )</td>
</tr>
<tr>
<td>Fallout</td>
<td>( \frac{FP}{FP + TN} )</td>
<td>( N'' )</td>
</tr>
<tr>
<td>Specificity</td>
<td>( \frac{TN}{TP + TN} )</td>
<td>( \frac{N' + N''}{N'} )</td>
</tr>
<tr>
<td>Negative predictive value</td>
<td></td>
<td>( \frac{N''}{N'} )</td>
</tr>
<tr>
<td>False discovery rate</td>
<td>( 1 - \text{precision} )</td>
<td>( 1 - \text{precision} )</td>
</tr>
<tr>
<td>False positive rate</td>
<td>( \frac{TN}{TP + TN} )</td>
<td>( 1 - \frac{N'}{N'} )</td>
</tr>
<tr>
<td>Likelihood ratio LR+</td>
<td>( \frac{1}{\text{sensitivity}} )</td>
<td>( \frac{1}{\text{sensitivity}} )</td>
</tr>
<tr>
<td>Likelihood ratio LR−</td>
<td>( \frac{1}{\text{sensitivity}} )</td>
<td>( \frac{1}{\text{sensitivity}} )</td>
</tr>
<tr>
<td>F-score (s = 0.5, 1.2)</td>
<td>( \frac{(TP + TN) - (FP + FN)}{TP + FN + TN + FP + FN} )</td>
<td>( 1 - \left( \frac{(TP + TN) - (FP + FN)}{TP + FN + TN + FP + FN} \right) )</td>
</tr>
<tr>
<td>Mathews correlation coefficient</td>
<td>( \sqrt{\left( \frac{TP + FN}{TP + FN + FP + TN} \right) \left( \frac{TP + TN}{TP + TN + FP + FN} \right)} )</td>
<td>( 1 - \left( \frac{TP + FN}{TP + FN + FP + TN} \right) \left( \frac{TP + TN}{TP + TN + FP + FN} \right) )</td>
</tr>
</tbody>
</table>
removing lanes. All the methods work relatively well with respect to FNs—that is, usually, the user does not need to add new lanes with these methods—the sum algorithm being the best one.

Figure 6 includes the representation of the accuracy, precision, sensitivity, specificity, negative predictive value and F-score of the four best lane-segmentation algorithms according to the data included in Table 5: average, derivative, STD and STD-derivative. Average is the best method in four of the six characteristics (accuracy, precision, specificity and F-score), and it is similar to STD in sensitivity and negative predictive value (the best in these characteristics). Average has a mean above 0.95 in all these six characteristics, and the second best method is STD with a mean above 0.89 in these characteristics.

When we study the lane-detection algorithms in the images grouped by their quality, we obtain a similar classification of the methods in the three categories (bad, intermediate and good quality). Average is the best method in the three categories. It has a mean of <1 FP in all the categories. With good images it obtains a mean of 0.17 FNs, which equals the best methods (sum and derivative) in that category, and obtains 1.1 FNs in the bad-quality images. See Supplementary Appendix C for details.

We focus now on the evaluation of lane-segmentation algorithms regarding the segmentation criterion. Table 6 includes the means and standard deviations of different methods to obtain the segmentation of the lanes. In this case, FPs and negatives are the number (divided by 1000) of pixels that should be removed or added to the segmentation, respectively. There exist significant differences among the methods in all the studied aspects. Derivative and STD-derivative are the methods with significant less FP but with more FNs. On the contrary, average is the method with significant less FNs but with more FPs. This property is shared with the rest of the methods: more FN (positive) follows less FP (negative).

Figure 7 includes the representation of the accuracy, precision, sensitivity, specificity, negative predictive value and F-score of the four best lane-segmentation algorithms according to the data included in Table 6: average, maximum, STD and sum. It is worth noting that the big amount of TN in the segmentation makes accuracy, specificity and predictive negative value almost 1 in all the methods. Average has the best sensitivity and F-score, and derivative and STD-derivative the best precision. Average and STD are the best methods with almost a mean above 0.85 in all these six characteristics.

Again, when we study the lane-segmentation algorithms in the images grouped by their quality (Supplementary Appendix C), we obtain a similar classification of the methods in the three categories (bad, intermediate and good quality).

In the literature, there are several papers that use the lane-detection measures included here to evaluate lane-segmentation algorithms; however, none of them evaluates how good is the actual segmentation of the lanes, or compares different approaches. The only paper that evaluates algorithms from different sources is [20], it takes two data sets of gel images and measures the performance of the algorithms presented in [20, 24, 67, 68]—all these algorithms use the average profile as core algorithm—considering the number of TP, FP and FN, and the recall, precision and F-score (s = 1) values; however, no
Table 6. Evaluation of surveyed profiles for lane segmentation

<table>
<thead>
<tr>
<th>Metric</th>
<th>AVG</th>
<th>BIN</th>
<th>DER</th>
<th>MAX</th>
<th>STD</th>
<th>STDDER</th>
<th>SUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>0.98 (0.01)</td>
<td>0.96 (0.02)</td>
<td>0.96 (0.02)</td>
<td>0.97 (0.02)</td>
<td>0.98 (0.02)</td>
<td>0.97 (0.02)</td>
<td>0.97 (0.02)</td>
</tr>
<tr>
<td>Precision</td>
<td>0.84 (0.09)</td>
<td>0.85 (0.09)</td>
<td>0.93 (0.08)</td>
<td>0.84 (0.09)</td>
<td>0.83 (0.1)</td>
<td>0.92 (0.08)</td>
<td>0.85 (0.07)</td>
</tr>
<tr>
<td>Sensitivity</td>
<td>0.92 (0.06)</td>
<td>0.66 (0.13)</td>
<td>0.46 (0.1)</td>
<td>0.73 (0.14)</td>
<td>0.85 (0.1)</td>
<td>0.62 (0.15)</td>
<td>0.78 (0.13)</td>
</tr>
<tr>
<td>Specificity</td>
<td>0.98 (0.01)</td>
<td>0.99 (0.01)</td>
<td>1 (0.01)</td>
<td>0.99 (0.01)</td>
<td>0.99 (0.01)</td>
<td>1 (0.01)</td>
<td>0.99 (0.01)</td>
</tr>
<tr>
<td>NPV</td>
<td>0.99 (0.01)</td>
<td>0.97 (0.03)</td>
<td>0.96 (0.02)</td>
<td>0.98 (0.02)</td>
<td>0.99 (0.01)</td>
<td>0.97 (0.02)</td>
<td>0.98 (0.02)</td>
</tr>
<tr>
<td>F-score</td>
<td>0.87 (0.06)</td>
<td>0.73 (0.1)</td>
<td>0.61 (0.1)</td>
<td>0.77 (0.1)</td>
<td>0.84 (0.08)</td>
<td>0.73 (0.13)</td>
<td>0.80 (0.09)</td>
</tr>
<tr>
<td>TP</td>
<td>66.71 (49.6)</td>
<td>46.76 (37.48)</td>
<td>11.38 (13.55)</td>
<td>55.98 (57.4)</td>
<td>63.72 (47.98)</td>
<td>17.58 (17.99)</td>
<td>50.75 (37.07)</td>
</tr>
<tr>
<td>FN</td>
<td>29.54 (29.66)</td>
<td>112.66 (72.48)</td>
<td>179.94 (96.3)</td>
<td>90.09 (61.4)</td>
<td>51.85 (40.4)</td>
<td>125.97 (72.7)</td>
<td>71.27 (54.66)</td>
</tr>
</tbody>
</table>

Note. **P < 0.001.
Top. Means (standard deviations) of different methods to obtain lane rois (N = 50). Bottom. Differences among methods and paired differences between two different methods after Bonferroni correction.

Figure 7. Representation of the accuracy, precision, specificity, sensitivity, negative predictive value and F-score of the four best lane-segmentation algorithms according to the data included in Table 6.

is the method with significant less FPs and the second method with less FNs (only slightly overrated by sum).

Figure 8 includes the representation of the AUROC, accuracy, precision, sensitivity, specificity, negative predictive value and F-score of the four best band-detection algorithms according to the data included in Table 5: average, maximum, STD and sum. AUROC can be considered as the fundamental comparative measure because it is independent of any threshold to differentiate between positive and negative samples, and it provides a good representation of the quality of the classifier in terms of both false-positive and false-negative detection [45]. Considering this parameter, Average and sum share the first position. Average is the best method in four of the other six characteristics (accuracy, precision, sensitivity and F-score), and sum has the best specificity and predictive negative value. Average is the best method, and together with maximum has a mean above 0.80 in all these six characteristics. STD-derivative (which is not included in the figure) has clearly the worst performance with most of the parameters below 0.5.

When we study the band-detection algorithms in the images grouped by their quality (see Supplementary Appendix C for details), we obtain a similar classification of the methods in the three categories (bad, intermediate and good quality). Average is the best method in the three categories. The FP means of the methods are stable with respect to the quality of the image. For instance, average only increases to a mean of 1.23 FPs with bad quality images. The false-negative means varies in one or two depending on the quality. For instance, average ranges from 2.31 with good images to 4.23 with bad images, or sum from 1.64 with good images to 3.23 with bad images.

In the literature, we can find a few papers that evaluate the performance of band-detection algorithms; nevertheless, none of these papers compares its approach with other methods. The performance of four software tools in [19] was measured using the number of TP, FP and FN. The algorithm presented in [10] was evaluated, with several configurations, in two sets of lane images with different qualities using as measures the TP rate (sensitivity), the FP rate (false-out) and the accuracy. The authors...
Table 7. Measures used in the literature to evaluate the performance of lane-segmentation and band-detection algorithms

<table>
<thead>
<tr>
<th>Authors</th>
<th>Data set size</th>
<th>Analyse</th>
<th>Evaluate</th>
<th>Measures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Akbari et al. [13]</td>
<td>20 lane images (456 lanes)</td>
<td>LD</td>
<td>Algorithms presented in the article</td>
<td>TP, FP, TN, SENS, SPC and ACC</td>
</tr>
<tr>
<td>Caricade et al. [19]</td>
<td>96 lane images, 12 gel images</td>
<td>BD, LD</td>
<td>Software tools</td>
<td>TP, FP and FN</td>
</tr>
<tr>
<td>Chan et al. [10]</td>
<td>2 sets of lane images</td>
<td>BD</td>
<td>Variants of the same algorithm</td>
<td>SENS, FPR and ACC</td>
</tr>
<tr>
<td>Ismail et al. [14]</td>
<td>20 lane images</td>
<td>LD</td>
<td>Variants of the same algorithm</td>
<td>FP and FN</td>
</tr>
<tr>
<td>Labed et al. [37]</td>
<td>15 lane images (63 bands)</td>
<td>BD</td>
<td>Variants of the same algorithm</td>
<td>TP, FP and FN</td>
</tr>
<tr>
<td>Lee et al. [27]</td>
<td>10 lane images (430 bands)</td>
<td>BD</td>
<td>Variants of the same algorithm</td>
<td>TP, FP and FN</td>
</tr>
<tr>
<td>Machado et al. [5]</td>
<td>22 gel images (365 lanes)</td>
<td>LD</td>
<td>Variants of the same algorithm</td>
<td>TP, FP and FN</td>
</tr>
<tr>
<td>Moreira et al. [20]</td>
<td>235 gel images (2073 lanes)</td>
<td>LD</td>
<td>Algorithms from several sources</td>
<td>TP, FN, TN, SENS, PPV and F1</td>
</tr>
<tr>
<td>Park et al. [32]</td>
<td>38 gel images</td>
<td>LD</td>
<td>Algorithm presented in the article</td>
<td>PPV, SENS and F1</td>
</tr>
<tr>
<td>Sotaquirá [36]</td>
<td>25 lane images, 25 gel images</td>
<td>BD, LD</td>
<td>Variants of the same algorithm</td>
<td>TP, FP and FN</td>
</tr>
<tr>
<td>Wong et al. [4]</td>
<td>161 gel images</td>
<td>LD</td>
<td>Software tools</td>
<td>SENS and SPC</td>
</tr>
</tbody>
</table>

Note. The following abbreviations are used:


Table 8. Evaluation of surveyed profiles

<table>
<thead>
<tr>
<th>Metric</th>
<th>AVG</th>
<th>BIN</th>
<th>DER</th>
<th>MAX</th>
<th>STD</th>
<th>STDDER</th>
<th>SUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUROC</td>
<td>0.89 (0.07)</td>
<td>0.77 (0.13)</td>
<td>0.8 (0.12)</td>
<td>0.87 (0.11)</td>
<td>0.86 (0.1)</td>
<td>0.44 (0.14)</td>
<td>0.89 (0.09)</td>
</tr>
<tr>
<td>Accuracy</td>
<td>0.88 (0.08)</td>
<td>0.75 (0.13)</td>
<td>0.71 (0.12)</td>
<td>0.81 (0.15)</td>
<td>0.84 (0.12)</td>
<td>0.46 (0.13)</td>
<td>0.8 (0.16)</td>
</tr>
<tr>
<td>Precision</td>
<td>0.93 (0.11)</td>
<td>0.85 (0.19)</td>
<td>0.79 (0.15)</td>
<td>0.81 (0.19)</td>
<td>0.87 (0.15)</td>
<td>0.42 (0.16)</td>
<td>0.78 (0.2)</td>
</tr>
<tr>
<td>Sensitivity</td>
<td>0.83 (0.12)</td>
<td>0.63 (0.19)</td>
<td>0.55 (0.21)</td>
<td>0.83 (0.14)</td>
<td>0.79 (0.14)</td>
<td>0.4 (0.16)</td>
<td>0.87 (0.11)</td>
</tr>
<tr>
<td>Specificity</td>
<td>0.93 (0.11)</td>
<td>0.88 (0.18)</td>
<td>0.87 (0.14)</td>
<td>0.81 (0.2)</td>
<td>0.89 (0.15)</td>
<td>0.52 (0.16)</td>
<td>0.77 (0.23)</td>
</tr>
<tr>
<td>NPV</td>
<td>0.86 (0.08)</td>
<td>0.74 (0.11)</td>
<td>0.69 (0.11)</td>
<td>0.85 (0.11)</td>
<td>0.83 (0.1)</td>
<td>0.5 (0.12)</td>
<td>0.88 (0.1)</td>
</tr>
<tr>
<td>F-score</td>
<td>0.87 (0.09)</td>
<td>0.71 (0.15)</td>
<td>0.63 (0.17)</td>
<td>0.80 (0.14)</td>
<td>0.82 (0.12)</td>
<td>0.4 (0.15)</td>
<td>0.81 (0.14)</td>
</tr>
<tr>
<td>FP</td>
<td>0.89 (1.25)</td>
<td>2.19 (4)</td>
<td>1.78 (1.25)</td>
<td>2.97 (3.03)</td>
<td>1.55 (1.68)</td>
<td>8.02 (2.85)</td>
<td>3.95 (4.36)</td>
</tr>
<tr>
<td>FN</td>
<td>2.72 (2.07)</td>
<td>5.57 (3.17)</td>
<td>7.21 (4.64)</td>
<td>2.78 (2.48)</td>
<td>3.28 (2.55)</td>
<td>9 (3.57)</td>
<td>2.06 (1.9)</td>
</tr>
</tbody>
</table>

Note. Differences after Bonferroni correction. F or χ²

**P < 0.001.

Top. Means (standard deviations) of different methods to obtain bands (N = 120). AUROC is computed using 25 successive thresholds for the height criterion. Bottom. Differences among methods and paired differences between two different methods after Bonferroni correction.

Discussion and conclusions

The analysis of DNA gel fingerprint images is a widely studied problem, and several approaches have been proposed to simplify the two stages that require user intervention: lane segmentation and band detection. In this article, we have surveyed the main techniques available in the literature—not only for lane segmentation and band detection but also for the other stages of the procedure. The conclusion for the surveyed lane-segmentation and band-detection algorithms is that, even if several approaches exist, they are all based on enhancing the location of peaks of a profile (it can be the average, binary,
derivative, maximum, STD, STD-derivative or sum profile) obtained from the horizontal (or vertical) projection of an image.

The examined papers rarely provide the implementation of their approaches, a comparison with other methods, or the data set that was used to evaluate their algorithms. Hence, our survey was not enough to establish what were the best algorithms for lane segmentation and band detection. We have overcome this drawback, thanks to the development of a publicly available benchmark that includes two data sets of gel and lane images, the gold standards associated with those images and a set of tools—implemented as ImageJ plug-ins—to analyse the performance of lane-segmentation and band-detection algorithms. The infrastructure provided by our benchmark avoids the burden of choosing data sets for testing algorithms, and reimplementing analysis tools and evaluation metrics for comparisons.

The evaluation measures implemented in the two plug-ins of the benchmark are related to three criteria: the correct segmentation of lanes, the correct detection of lanes and the correct detection of bands. Measures related to the latter two criteria have been previously used in the literature, see Table 7; however, the evaluation of lane-segmentation algorithms based on the correctness of the segmentation is new. None of the papers including an evaluation provides the data set, the tools to perform the measurements or the comparison among different algorithms—three aspects covered in the current article.

Finally, we have implemented the seven core algorithms, based on finding the location of peaks of seven profiles (average, binary, derivative, maximum, STD, STD-derivative and sum), to segment lanes and detect bands. These algorithms have been implemented inside ImageJ plug-ins and have been evaluated using our benchmark. In general, the average algorithm excels the rest of the algorithms regarding the three evaluated criteria: lane segmentation and lane and band detection.

The ImageJ plug-ins for lane segmentation and band detection are freely available and open source; therefore, they can be improved with the enhancements available in the literature. It is also worth mentioning that these plug-ins are a good starting point for fast prototyping because they can be combined with the vast number of features to process images available in ImageJ (e.g. filtering, background subtraction, morphological operations and so on); namely, they have been used as a basis for the GelJ tool [69]. Moreover, the plug-ins devoted to evaluate the performance of lane-segmentation and band-detection algorithms can be used in other contexts where the analysis of image segmentation or object detection is required. As further work, and as we have explained in ‘Similarity matrices’ section, it would be interesting to include in GelJ the texture-based method presented in [45] and compare it with the traditional band- and curve-based approaches for fingerprint comparisons.

Finally, and although it is out of the reach of this article, it is worth mentioning the explosive growth of genomic sequences. Such an impressive work provides an unprecedented opportunity to explore genetic variability and biological function of organisms from a fundamental point. In this line, there has recently been a rapid advance in developing various powerful web servers to formulate DNA/RNA sequences with their feature vectors [70–75].

**Key Points**
- We survey the techniques applied in DNA fingerprint analysis.
- We have created a benchmark for lane-segmentation and band-detection algorithms.
- Benchmark contains a data set of images and their gold standards.
- Tools to measure the performance of algorithms have been implemented in ImageJ.
- Core algorithms have been implemented in ImageJ, and tested using the benchmark.

**Supplementary Data**

**Supplementary data** are available online at https://academic.oup.com/bib.

**Funding**

This work was partially supported by the Agencia de Desarrollo Económico de La Rioja (ADER) [2013-I-001] and Ministerio de Economía y Competitividad [MTM2014-54151-P].

**References**


