# Model-Based Similarity Scores for the Comparison of Cartridge Case Impressions

Joseph Zemmels,[1] Heike Hofmann,[1] and Susan VanderPlas[2]
[1)] *Iowa State University*[a)]
[2)] *University of Nebraska - Lincoln*

(Dated: 26 September 2023)

Put your abstract here.

 [https://doi.org(DOI number)]

[XYZ] Pages: 1–13

## I. BACKGROUND & INTRODUCTION

A *cartride case* is the part of firearm ammunition that houses the projectile and propulsive device. When a firearm is discharged and the projectile travels down the barrel, the cartridge case moves in the opposite direction and slams against the back wall, the *breech face*, of the firearm. Markings on the breech face are "stamped" into the surface of the cartridge case leaving so-called *breech face impressions*. In this paper, we introduce an automatic method for measuring the similarity between two cartridge cases based on their breech face impressions.

### A. Traditional Cartridge Case Comparison

In a traditional examination, forensic examiners use these impressions analogous to a fingerprint to determine whether two cartridge cases were fired from the same firearm. First, two cartridge cases are collected - perhaps one is from a crime scene and the other is collected from a suspect's gun. An examiner places the two cartridge cases beneath a "comparison microscope" that merges the views of two compound microscopes into a single split view (Thompson, 2017). The examiner assesses the "degree of similarity" between the markings on the cartridge cases and reaches either an *identification*, meaning the cartridge cases were fired from the same firearm, an *elimination*, meaning they were fired from different firearms, or an *inconclusive*, meaning the evidence is insufficient to make an identification or elimination (AFTE Criteria for Identification Committee, 1992).[1]

Critics of traditional forensic examinations cite a lack of "foundational validity" underlying the procedures used by firearm and toolmark examiners (National Research Council, 2009; President's Council of Advisors on Science & Technology, 2016). In particular, examiners rely largely on their subjective findings rather than on a well-defined procedure to measure similarity. PCAST (2016) pushed for "developing and testing image-analysis algorithms" to objectively measure the similarity between cartridge cases.

In this paper, we introduce a procedure to automatically compare digital scans of cartridge cases. Throughout this paper, we use scans, taken by us [**scientific data citation**], of cartridge cases collected by Baldwin *et al.* (2014). The cartridge case scans are available as part of the data repository at [**doi here**] [**citation**]. We also provide code to reproduce the results shared in this paper at [**github link here**].

### B. Algorithmic Cartridge Case Comparisons

#### *1. Cartridge Case Surface Scans*

We captured digital representations of cartridge case surfaces using topographic scanning technology. The scanner measures the relative surface depth and stores these measurements in a 2D array called a *surface matrix*. The left side of Figure 1 depicts a surface matrix representing the region at the base of a cartridge case surface called the *primer*, which is the circular metal cap struck by the firing pin to initiate the firing process. The purple ring near the edges of the scan represent the boundary of the cartridge case primer while the darker orange ring near the center of the scan represents the deformation of metal caused by the contact with the firing pin.

Of particular interest is the annular breech face impression region around the firing pin impression. We isolate this region by applying a series of manual and automatic pre-processing steps to the surface matrix, resulting in the scan on the right side of Figure 1. The gray pixels in this image represent structurally missing values introduced during pre-processing. See [**scientific data citation**] for more information on the pre-processing procedure.

Two pre-processed cartridge cases are compared to measure the similarity of their breech face impressions. In the next section, we summarize a technique for comparing cartridge case scans called the *Congruent Matching Cells* algorithm (Song, 2013).

#### *2. Congruent Matching Cells Algorithm*

Recent proposals for automatic cartridge case scoring algorithms borrow from image processing and computer vision techniques. For example, Vorburger *et al.* (2007)

[a)] jzemmels@iastate.edu; other info

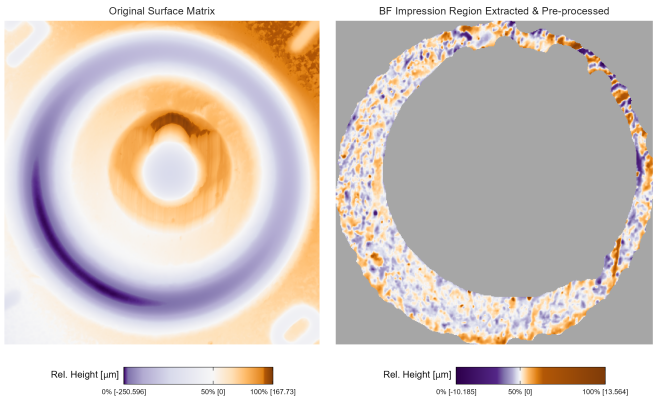


FIG. 1. We apply a sequence of pre-processing functions to each scan. Each pre-processing step further emphasizes the breech face impressions in the scan.

proposed using the cross-correlation function (CCF) to compare images or scans of cartridge case surfaces. The CCF measures the similarity between two matrices for all possible translations of one matrix against the other. Calculating the CCF while rotating one of the scans therefore allows for estimation of the optimal translation and rotation, together referred to as the *registration*, between the two scans; simply choose the rotation/translation at which the CCF is maximized.

Song (2013) noted that two matching cartridge cases often share similar impressions in specific regions, so calculating the CCF between two full scans may not highlight their similarities. Instead, Song (2013) proposed partitioning one cartridge case scan into a grid of "cells" and calculating the CCF between each cell and the other scan. If two cartridge cases are truly matching, then the maximum CCF value between each cell and the other scan, particularly the cells containing distinguishable breech face impressions, should be relatively large.

Furthermore, the cells should "agree" on the registration at which the CCF is maximized. **[Visually, this corresponds to xyz — borrow figure from scientific data paper]** Song (2013) outlined the "Congruent Matching Cells" algorithm to determine the number of cells that agree on a particular registration. A cell is classified as a Congruent Matching Cell (CMC) if its estimated registration is within some threshold of the median registration across all cells and its CCF value is above some threshold (see C for more details). A number of follow-up papers proposed alterations to the the original CMC method (Chen *et al.*, 2017; Tong *et al.*, 2015). Zemmels *et al.* (2022) introduced an open-source implementation of the CMC method in the `cmcR` R package. As an alternative to defining Congruent Matching Cells, Zhang *et al.* (2021) proposed using a clustering algorithm from Ester *et al.* (1996) to determine the number of cells in agreement on a specific registration.

The underlying CMC criteria are a set of binary rules; for example, a cell's associated registration either is or is not within a pre-defined threshold of the consensus-estimated registration. While interpretable, these threshold-based rules are quite sensitive to the choice of threshold as demonstrated in Zemmels *et al.* (2023). We propose a more robust, model-based method that relies on numerical features to measure the strength of consensus rather than binary criteria. We also introduce a novel cross-validation procedure to learn and test optimal parameters for this cartridge case algorithm.

## II. METHODS

### A. Notational Conventions

First, we establish notation that will be used to define the features. We introduce additional notation in subsequent sections as it becomes relevant. Let $A$ and $B$ denote two surfaces matrices that we wish to compare. For simplicity, we assume $A, B \in \mathbb{R}^{k\times k}$ for a positive integer $k$.[2] We use lowercase letters and subscripts to denote a particular value of a matrix: $a_{ij}$ is the value in the $i$-th row and $j$-th column, indexed starting from the top-left corner, of matrix $A$.

To accommodate structurally missing values, we adapt standard matrix algebra by encoding the notion of "missingness" into the space of real values as follows: if an element of either matrix $A$ or $B$ is missing, then any element-wise operation including this element is also missing. Standard matrix algebra holds for non-missing elements. For example, the addition operator is defined as:

$$A \oplus_{NA} B = (a_{ij} \oplus_{NA} b_{ij})_{1\le i,j\le k} = \begin{cases} a_{ij} + b_{ij} & \text{if both } a_{ij} \text{ and } b_{ij} \text{ are numbers} \\ NA & \text{otherwise} \end{cases}$$

Other element-wise operations such as $\ominus_{NA}$ are defined similarly. For readability, we will use standard operator notation $+, -, >, <, I(\cdot), \ldots$ and assume the extended, element-wise operations as defined above. Note that this definition of dealing with missing values is consistent with a setting of `na.rm = FALSE` in terms of calculations in R (R Core Team, 2017).

We call cartridge cases that originated from the same firearm "matches" and those that originated from different firearms "non-matches." In the following sections, we use the two known-match cartridge cases in Figure 2 as example matrices $A$ and $B$.

### B. Registration Estimation

A critical step in comparing $A$ and $B$ is to find a transformation of $B$ such that it aligns best to $A$ (or vice versa). In image processing, this is called *image registration.* Noting that $A$ and $B$ are essentially grayscale images with structurally missing values, we rely on a standard image registration technique (Brown, 1992).

In our application, a registration is composed of a discrete translation by $(m, n) \in \mathbb{Z}^2$ and rotation by

$\theta \in [-180°, 180°]$. To determine the optimal registration, we calculate the *cross-correlation function* (CCF) between $A$ and $B$, denoted $(A \star B)$, which measures the similarity between $A$ and $B$ for every possible translation of $B$. We estimate the registration by calculating the maximum CCF value across a range of rotations of matrix $B$. Let $B_\theta$ denote $B$ rotated by an angle $\theta \in [-180°, 180°]$ and $b_{\theta_{mn}}$ the $m,n$-th element of $B_\theta$. Then the estimated registration $(m^*, n^*, \theta^*)$ is:

$$(m^*, n^*, \theta^*) = \arg\max_{m,n,\theta}(a \star b_\theta)_{mn}.$$

In practice we consider a discrete grid of rotations $\boldsymbol{\Theta} \subset [-180°, 180°]$. The registration procedure is outlined in algorithm 1. We refer to the matrix that is rotated as the "target." The result is the estimated registration of the target matrix to the "source" matrix.

**Algorithm 1:** Image Registration Procedure

**Data:** Source matrix $A$, target matrix $B$, and rotation grid $\boldsymbol{\Theta}$

**Result:** Estimated registration of $B$ to $A$, $(m^*, n^*, \theta^*)$, and cross-correlation function maximum, $CCF_{\max}$

**for** $\theta \in \boldsymbol{\Theta}$ **do**
- Rotate $B$ by $\theta$ to obtain $B_\theta$;
- Calculate $CCF_{\max,\theta} = \max_{m,n}(a \star b_\theta)_{mn}$;
- Calculate translation $[m^*_\theta, n^*_\theta] = \arg\max_{m,n}(a \star b_\theta)_{mn}$

**end**

Calculate overall maximum correlation $CCF_{\max} = \max_\theta\{CCF_{\max,\theta} : \theta \in \boldsymbol{\Theta}\}$;

Calculate rotation $\theta^* = \arg\max_\theta\{CCF_{\max,\theta} : \theta \in \boldsymbol{\Theta}\}$;

**return** *Estimated rotation* $\theta^*$*, translation* $m^* = m^*_{\theta^*}$ *and* $n^* = n^*_{\theta^*}$*, and* $CCF_{\max}$

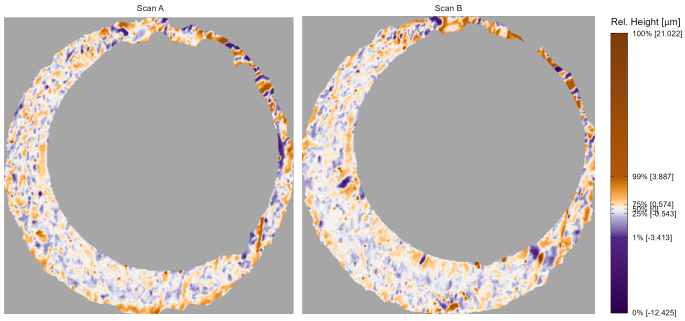


FIG. 2. A matching pair of processed cartridge case scans. We measure the similarity between these cartridge cases using the distinguishable breech face impressions on their surfaces.

Note that the calculation of the CCF requires that all missing values, including structural missing values, are imputed in $A$ and $B$. We impute missing values in a scan with the average non-missing value in that scan. As a result of imputing a large number of missing values, we found in our experimentation the estimated registrations $(\theta^*, m^*, n^*)$ to be reliable but the value of $CCF_{\max}$ to not be a reliable measure of similarity for scans. We discuss how we compute more reliable measures of similarity in II C.

### 1. Full-Scan Registration

We first estimate the registration between two full scans $A$ and $B$ using algorithm 1 with a rotation grid $\boldsymbol{\Theta} = \{-30°, -27°, ..., 27°, 30°\}$. This results in an estimated registration $(m^*, n^*, \theta^*)$ and similarity measure $CCF_{\max}$. We also perform algorithm 1 with the roles of $A$ and $B$ reversed, meaning the target scan $A$ is aligned to source scan $B$.

To accommodate these two comparison directions, we introduce a new subscript $d = A, B$, referring to the source scan in algorithm 1. Consequently, we obtain two sets of estimated registrations, $(m^*_d, n^*_d, \theta^*_d)$ and $CCF_{\max,d}$, for $d = A, B$.[3]

### 2. Cell-Based Registration

We next perform a cell-based comparison procedure, which begins with selecting one of the matrices, say $A$, as the "source" matrix that is partitioned into a grid of cells. The left side of Figure 3 shows an example of such a cell grid overlaid on a scan. Each of these source cells will be compared to the "target" matrix, in this case $B^*$. Because $A$ and $B^*$ are already partially aligned from the full-scan registration procedure, we compare each source cell to $B^*$ using a new rotation grid of $\boldsymbol{\Theta}'_A = \{\theta^*_A - 2°, \theta^*_A - 1°, \theta^*_A, \theta^*_A + 1°, \theta^*_A + 2°\}$.

We now extend the surface matrix notation introduced previously to accommodate cells. Let $A_t$ denote the $t$-th cell of matrix $A$, $t = 1, ..., T_A$ where $T_A$ is the total number of cells containing non-missing values in scan $A$ (e.g., $T_A = 43$ in Figure 3) and let $(a_t)_{ij}$ denote the $i,j$-th element of $A_t$. The cell-based comparison procedure is outlined in algorithm 2.

**Algorithm 2:** Cell-Based Comparison Procedure

**Data:** Source matrix $A$, target matrix $B^*$, grid size $R \times C$, and rotation grid $\boldsymbol{\Theta}'_A$

**Result:** Estimated translations and $CCF_{\max}$ values per cell, per rotation

Partition $A$ into a grid of $R \times C$ cells;

Discard cells containing only missing values, leaving $T_A$ remaining cells;

**for** $\theta \in \boldsymbol{\Theta}'_A$ **do**
- Rotate $B^*$ by $\theta$ to obtain $B^*_\theta$;
- **for** $t = 1, ..., T_A$ **do**
  - Calculate $CCF_{\max,A,t,\theta} = \max_{m,n}(a_t \star b^*_\theta)_{mn}$;
  - Calculate translation $[m^*_{A,t,\theta}, n^*_{A,t,\theta}] = \arg\max_{m,n}(a_t \star b^*_\theta)_{mn}$
- **end**

**end**

**return** $\boldsymbol{F}_A = \{(\theta, m^*_{A,t,\theta}, n^*_{A,t,\theta}, CCF_{\max,A,t,\theta}) : \theta \in \boldsymbol{\Theta}'_A, t = 1, ..., T_A\}$

We can think of algorithm 1 as a specific case of algorithm 2 where $R = C = 1$, but we distinguish the algorithms in this paper based on the intended use of the result. Rather than exclusively returning the registration that maximizes the overall CCF as in algorithm 1, algorithm 2 returns the set $\boldsymbol{F}_A$ of translations and CCF values for each of the $T_A$ cells and each rotation in $\boldsymbol{\Theta}'_A$.

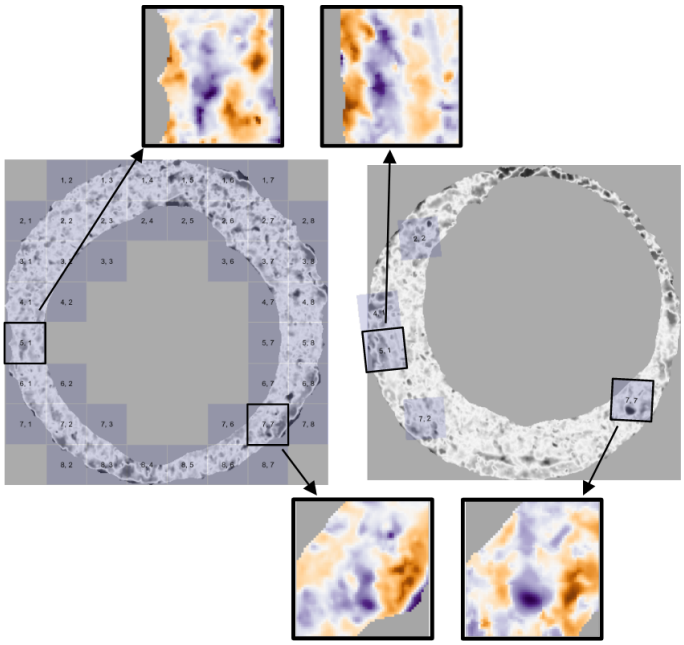

FIG. 3. Estimated registrations of cells from a non-match pair of cartridge cases. A source scan (left) is separated into an $8 \times 8$ grid of cells. We exclude cells containing only missing values (visualized here as gray pixels). Each source cell is compared to a target scan (right) to estimate where it aligns best. We show a handful of cells at their estimated alignment in the target scan and magnify the surfaces captured by cell pairs 5, 1 and 7, 7. Although the cartridge case pair is non-matching, we note that there are similarities in the surface markings for these cell pairs.

Figure 3 shows the estimated registrations of cells between two non-match cartridge cases. We magnify the surface values captured by cell pairs 5, 1 and 7, 7 and note the similarities in the surface values; for example, the dark purple region in the middle of the cell 7, 7 pair.

Next, we introduce a set of similarity features for two cartridge case scans. We calculate features at two scales: between two full scans and between individual cells. Analogous to how a forensic examiner uses a comparison microscope with different magnification levels, this allows us to assess the similarity between two scans at the macro and micro levels.

## C. Feature Calculation

The result of the full scan and cell-based comparisons is a set of estimated registrations with associated $CCF_{\max}$ values. We compute statistics on these results based on how we expect these registrations to behave for matching vs. non-matching cartridge case pairs. Similar to the assumption made in the Congruent Matching Cells algorithm, we expect that cells will agree on a particular registration if the two cartridge cases truly match, suggesting that a measure of spread of the estimated registrations would be an informative statistic. Further, we would expect that the consensus-estimated registrations should be approximately opposite across the two comparison directions (i.e., registering $A$ to $B$ is the opposite of registering $B$ to $A$).

We first introduce a set of *registration-based features* based on descriptive statistics of the full scan and cell-based registrations. Then, we discuss a set of *density-based features* based on applying a density-based clustering algorithm to the cell registrations $\boldsymbol{F}_A$ and $\boldsymbol{F}_B$.

### *1. Registration-based Features*

For $d = A$, we apply the registration transformation $(m^*_A, n^*_A, \theta^*_A)$ to $B$ to obtain $B^*$. Because the calculation of the $CCF_{\max}$ requires imputing missing values in both the source and target scans, we found the $CCF_{\max}$ values to not be a reliable measure of similarity for two cartridge cases. Instead, we compute the *pairwise-complete correlation*, *cor*, which is equivalent to the Pearson correlation between the overlapping, non-missing elements of $A$ and $B^*$.

We compute the pairwise-complete correlation between $A$ and $B^*$, resulting in $cor_{\text{full},A}$. We repeat this in the other comparison direction to obtain $cor_{\text{full},B}$ and average the two: $cor_{\text{full}} = \frac{1}{2}(cor_{A,\text{full}} + cor_{B,\text{full}})$. We assume that the **full-scan pairwise-complete correlation**, $cor_{\text{full}}$, is large for truly matching cartridge cases.

Just as with the whole-scan registration, we calculate the pairwise-complete correlation between each cell $A_t$ and a matrix $B^*_{\theta,t}$ of the same size extracted from $B^*_\theta$ after translating by $[m^*_{A,\theta}, n^*_{A,\theta}]$. From this we obtain a set of pairwise-complete correlations for each cell and rotation: $\{cor_{A,t,\theta} : t = 1, ..., T_A, \theta \in \boldsymbol{\Theta}'_A\}$.

We repeat algorithm 2 and the pairwise-complete correlation calculation using $B$ as the source scan and $A^*$ as the target, resulting in cell-based registration set $\boldsymbol{F}_B$ and pairwise-complete correlations $\{cor_{B,t,\theta} : t = 1, ..., T_B, \theta \in \boldsymbol{\Theta}'_B\}$.

For $d = A, B$ and $t = 1, ..., T_d$, define the cell-wise maximum pairwise-complete correlation as:

$$cor_{d,t} = \max_\theta \{cor_{d,t,\theta} : \theta \in \boldsymbol{\Theta}'_d\}.$$

We compute two features, the **average** and **standard deviation of the cell-based pairwise-complete correlations**, using the correlation data:

$$\overline{cor}_{\text{cell}} = \frac{1}{T_A + T_B} \sum_{d \in \{A,B\}} \sum_{t=1}^{T_d} cor_{d,t}$$

$$s_{cor} = \sqrt{\frac{1}{T_A + T_B - 1} \sum_{d \in \{A,B\}} \sum_{t=1}^{T_d} (cor_{d,t} - \overline{cor}_{\text{cell}})^2}.$$

We expect $\overline{cor}_{\text{cell}}$ and $s_{cor}$ to be large for truly matching cartridge case pairs relative to non-matching pairs.

For $d = A, B$ and $t = 1, ..., T_d$, define the per-cell estimated translations and rotation as:

$$\theta^*_{d,t} = \arg\max_{\theta} \{CCF_{\max,d,t,\theta} : \theta \in \mathbf{\Theta}'_d\}$$
$$m^*_{d,t} = m^*_{\theta^*_{d,t},d,t}$$
$$n^*_{d,t} = n^*_{\theta^*_{d,t},d,t}.$$

We compute the **standard deviation of the cell-based estimated registrations** using the estimated translations and rotations:

$$s_{\theta^*} = \sqrt{\frac{1}{T_A + T_B - 1} \sum_{d \in \{A,B\}} \sum_{t=1}^{T_d} (\theta^*_{d,t} - \bar{\theta}^*)^2}$$

$$s_{m^*} = \sqrt{\frac{1}{T_A + T_B - 1} \sum_{d \in \{A,B\}} \sum_{t=1}^{T_d} (m^*_{d,t} - \bar{m}^*)^2}$$

$$s_{n^*} = \sqrt{\frac{1}{T_A + T_B - 1} \sum_{d \in \{A,B\}} \sum_{t=1}^{T_d} (n^*_{d,t} - \bar{n}^*)^2}$$

where

$$\bar{m}^* = \frac{1}{T_A + T_B} \sum_{d \in \{A,B\}} \sum_{t=1}^{T_d} m^*_{d,t}$$

$$\bar{n}^* = \frac{1}{T_A + T_B} \sum_{d \in \{A,B\}} \sum_{t=1}^{T_d} n^*_{d,t}$$

$$\bar{\theta}^* = \frac{1}{T_A + T_B} \sum_{d \in \{A,B\}} \sum_{t=1}^{T_d} \theta^*_{d,t}.$$

We expect $s_{\theta^*}, s_{m^*}, s_{n^*}$ to be small for truly matching cartridge case pairs relative to non-matching pairs.

### 2. Density-Based Features

We wish to identify when multiple cells agree on, or cluster around, a particular registration value. However, pursuant with the notion that only certain regions of matching cartridge cases contain distinctive markings, it is unreasonable to assume and empirically rare that **all** cells agree on a single registration. For example, the left scatterplot in Figure 4 shows the per-cell estimated translations $[m^*_{A,t,\theta}, n^*_{A,t,\theta}]$ when scan $A$ is used as source and $B^*$ as target rotated by $\theta = 3°$. The right scatterplot shows the per-cell estimated translations with the roles of $A$ and $B^*$ reversed for $\theta = -3°$. We see distinctive clusters, the black points, in both plots among many noisy, gray points. The task is to isolate the clusters among such noise.

We use the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm proposed by Ester *et al.* (1996) to identify clusters. Compared to other clustering algorithms such as k-means (MacQueen, 1967), DBSCAN does not require a pre-defined number of expected clusters. Instead, the algorithm forms clusters if the number of points within an $\epsilon > 0$ distance of a point exceeds some pre-defined threshold, $minPts > 1$. If a point does not belong to a cluster, then DBSCAN labels that point as "noise."

In Figure 4, we use a $4 \times 4$ grid of cells to which we apply DBSCAN with $\epsilon = 3$ and $minPts = 3$. The resulting clusters for the two comparison directions are 7 and 9 cells, respectively, visualized in Figure 4 as black points. This indicates that about half of the cells in the $4 \times 4$ grid agree on a registration in both comparison directions. Additionally, the mean cluster centers are approximately close to 0: $(\hat{m}_A, \hat{n}_A, \hat{\theta}_A) \approx (0.29, 0.57, 0°)$ when $A$ is used as source compared to $(\hat{m}_B, \hat{n}_B, \hat{\theta}_B) \approx (-0.67, 0.00, 0°)$ when $B^*$ is used as source. If $A$ and $B$ were truly matching and the full scan registration from algorithm 1 successfully aligned the two scans, then we wouldn't expect the cells to move much in their respective registrations, which is illustrated in this example. For non-matching scans, we wouldn't expect the cell registrations to agree only by coincidence.

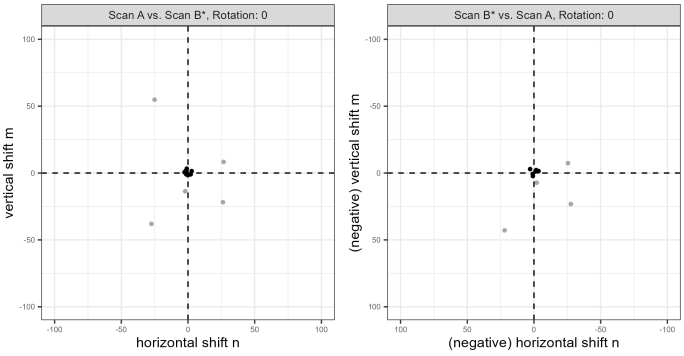


FIG. 4. Cluster assignments based on the Density Based Spatial Clustering with Applications to Noise (DBSCAN) algorithm for estimated translations in two comparison directions. Clustered points are shown as black while "noise" points are gray. We see that the clusters are both centered near the origin, indicating no further transformation is needed. Points are jittered for visibility.

To calculate the density-based features, we first use a 2D kernel density estimator (Venables and Ripley, 2002)

to identify the rotation $\hat{\theta}_d$ at which the per-cell translations achieve the highest density. Next, we compute clusters using the DBSCAN algorithm amongst the estimated translations $\{(m^*_{d,t,\hat{\theta}_d}, n^*_{d,t,\hat{\theta}_d}) : t = 1, ..., T_d\}$ like those shown in Figure 4.[4] Let $\boldsymbol{C}_d$ denote the set of cells in the DBSCAN cluster. We treat the mean cluster centers as the estimated translations $[\hat{m}_d, \hat{n}_d]$.

We calculate four features from the density-based clustering procedure: **average DBSCAN cluster size** $C$, the **DBSCAN cluster indicator** $C_0$, and the **root sum of squares of the dens**ity-estimated registrations $(\Delta_\theta, \Delta_{\text{trans}})$ defined as:

$$
\begin{aligned}
C &= \frac{1}{2}\left(|\boldsymbol{C}_A| + |\boldsymbol{C}_B|\right) \\
C_0 &= I(|\boldsymbol{C}_A| > 0 \text{ and } |\boldsymbol{C}_B| > 0) \\
\Delta_\theta &= |\hat{\theta}_A + \hat{\theta}_B| \\
\Delta_{\text{trans}} &= \sqrt{(\hat{m}_A + \hat{m}_B)^2 + (\hat{n}_A + \hat{n}_B)^2}
\end{aligned}
$$

where $|\boldsymbol{C}_d|$ denotes the cardinality of $\boldsymbol{C}_d$ and $I(\cdot)$ is the identity function equal to 1 if the predicate argument "$\cdot$" evaluates to TRUE and 0 otherwise. We use both $C$ and $C_0$ because of potential missingness in the values of $C$ if no cluster is identified. Missing $C$ values are imputed using the median non-missing value when fitting classifiers, so the missingness information is retained in $C_0$.

For truly matching cartridge case pairs, we expect $C$ to be large and $\Delta_\theta, \Delta_{\text{trans}}$ to be small relative to non-matching pairs and for $C_0$ to be equal to 1.

In summary, there are 10 total features, 6 registration-based and 4 density-based, that we calculate for each cartridge case pair. We list the 10 features in Table I. In the next section, we discuss how we use these features in a model-based approach to obtain similarity scores.

| | |
|---|---|
| $cor_{\text{full}} \in [0, 1]$ | Full-scan pairwise-complete correlation |
| $\overline{cor}_{\text{cell}} \in [0, 1]$ | Average cell-based pairwise-complete correlation |
| $s_{cor} \in [0, \infty)$ | Standard deviation of the cell-based pairwise-complete correlations |
| $s_{m^*} \in [0, \infty)$ | Standard deviation of the cell-based vertical translations (in microns) |
| $s_{n^*} \in [0, \infty)$ | Standard deviation of the cell-based horizontal translations (in microns) |
| $s_{\theta^*} \in [0, \infty)$ | Standard deviation of the cell-based rotations (degrees) |
| $C \in \{minPts, ..., T_S\}$ | Average DBSCAN cluster size (for scan $S$) |
| $C_0 \in \{0, 1\}$ | DBSCAN cluster indicator |
| $\Delta_\theta \in [0°, 180°)$ | Absolute sum of the density-estimated rotations (degrees) |
| $\Delta_{\text{trans}} \in [0, \infty)$ | Root sum of squares of the density-estimated translations (in microns) |

TABLE I. Ten features we compute for each cartridge case pair.

## D. Model Fitting

We use a data set of 510 cartridge cases scanned from 25 firearms. We randomly split the data into 10 firearms for training and 15 firearms for testing. This resulted in a training data set of 210 cartridge cases, $\binom{210}{2} = 21,945$ pairwise comparisons, and a testing set of 300 cartridge cases, $\binom{300}{2} = 44,850$ pairwise comparisons. Because we consider every pairwise comparison between these scans, there is a relatively large class imbalance between matches and non-matches in these data sets. Specifically, non-matching comparisons make up 19,756 of the 21,945 (90.0%) training comparisons and 41,769 of the 44,850 (93.1%) testing comparisons.

We use 10-fold cross-validation repeated thrice (Kuhn, 2022) to train two binary classifiers based on a logistic regression and a random forest (Breiman, 2001; Liaw and Wiener, 2002). These models predict the probability that a pair of cartridge cases match (the *match probability*). Then, the model classifies the pair as a match or non-match depending on whether the match probability exceeds a set threshold.

Models trained to maximize accuracy on imbalanced data often exhibit a "preference" for classifying new observations as the majority class (Fernández *et al.*, 2018), which in our case are non-matches. To address this imbalance, we explore three possible subsampling techniques: *upsampling* the number of match comparisons to equal the number of non-match comparisons, *downsampling* the number of non-match comparisons to equal the number of match comparisons, and performing no subsampling.

An optimization criterion commonly used for imbalanced data is to select the model that maximizes the area under the Receiver Operating Characteristic (ROC) curve, which measures the performance of a model under different threshold values (James *et al.*, 2013). The model that maximizes this area, commonly abbreviated AUC, is one that performs best under a variety of threshold values relative to the other models - this consistency is a desired trait. Using the ROC curve, we choose the match probability threshold that balances the true negative and true positive (equivalently, the false positive and false negative) rates on the training data.

We optimize the models by performing a grid search across the following parameters:

- DBSCAN parameters $\epsilon \in \{3, 4, ..., 15\}$ and $minPts \in \{3, 4, ..., 10\}$,
- Subsampling technique: downsampling, upsamplng, and no subsampling,

- For the random forest, the "mtry" variable, which controls the number of candidate variables a decision tree has available at each split (Breiman, 2001), and
- Match probability, $p \in [0, 1]$

Once we have a trained model, we use it to predict the match probability and classify a new cartridge case pair. We use this match probability as a similarity score where larger values correspond with more similar cartridge cases. We compute this score for the pairwise comparisons in the test data as a means of comparing the generalizability of the various models. The following section details the results of this cross-validation training/testing procedure. We refer the reader to **[link]** for the source code used to derive these results.

## III. RESULTS

### A. ROC Curves

First, we consider results from the training procedure. Figure 5 shows the resulting ROC curves for three classifiers trained on the training data set. For comparison to the logistic regression (LR) and random forest (RF) classifiers, we also consider a classifier based on the Congruent Matching Cells (CMC) method proposed in Song (2013). We obtained these CMC results using the implementation available in the `cmcR` R package (Zemmels *et al.*, 2022) and the same training and testing set. Similar to the two statistical models, we chose CMC method parameters using the AUC as an optimization criterion. We note that the interpretation of the thresholds used for the LR and RF models, which classify a pair as a match if its predicted class probability exceeds a threshold, differ from that of the CMC method, which classifies a pair as a match if its associated CMC count exceeds a threshold.

The ROC curves allow us to visually compare the behavior of these three classifiers under various score thresholds where curves closer to the top-left corner are preferred. The LR and RF models perform comparably as evidenced by the similar curves on the right side of Figure 5. The left side shows a zoomed-in version of the top left corner of plot, where we are more easily able to identify the superior performance of random forest model. The CMC method performs comparably worse as evidenced by the noticeably lower ROC curve.

To numerically compare the three classifiers, we compute the area under the ROC curve (AUC) as well as the score threshold (Thresh.) that balances the false negative and false positive rates (the equal error rate or EER). The AUC for logistic regression and random forest classifiers are much higher than the AUC of the CMC method classifier. Each model has a different score threshold that yields the equal error rate, which we visualize as points along the three ROC curves in Figure 5. We use these thresholds to compute both the training and test classification results summarized below. We see that the random forest classifier has the lowest equal error rate out of the three models with the logistic regression classifier a close second.

### B. Optimized Model Comparison

Figure 6 summarizes the training and testing accuracy, true negative and true positive rates for four binary classifiers. We distinguish between the training and testing results using gray and black points/line segments, respectively, which allows us to assess the generalizability of the various models to new comparisons. The conclusions drawn from Figure 6 are intended to primarily be qualitative and comparative across models. Table IV and Table V in the Appendix provide a numerical summary of these results.

We first compare the training and testing results across the five models and three columns in Figure 6. In general, the true negative rates based on the test data are slightly lower than those of the training data indicating that the models' ability to distinguish between non-matching comparisons generalizes well to the testing data. In contrast, the true positive rates tend to be lower for the test data compared to the training data across the various models, which indicates a potential difference between the training and testing data. As we discuss below, there is a single firearm among the 15 test firearms that contributes the majority of false negative (misclassified match) test classifications. Despite lower true positive rates, the overall accuracy between the training and testing sets are comparable due to the large class imbalance between matching and non-matching comparisons in both.

In the first row, we consider a baseline classifier based solely on the Cluster Indicator feature $C_0$. Namely, if the DBSCAN algorithm finds clusters in the cell-based translations from both directions of a cartridge case comparison, then that pair is classified as a match. This is analogous to the classification rule used in Zhang *et al.* (2021). We optimized this $C_0$-based classifier by choosing the DBSCAN parameters $\epsilon$ and $minPts$ that resulted in the most balanced training true negative and true positive rates, resulting in $\epsilon = 15$ and $minPts = 8$. The optimized $C_0$-based classifier performs considerably worse across the three measures compared to the other models with test accuracy 89.44%, true negative rate 89.64%, and true positive rate 86.82%.

The second row of Figure 6 summarizes results from training the two classifier models on a subset of the full ACES feature set consisting of the Cluster Indicator feature $C_0$ and the six registration-based features summarized in **??**. We consider this subset of features to represent the features used in Congruent Matching Cells methods (Song, 2013; Zhang *et al.*, 2021). In general, we see that the logistic regression (LR) and random forest (RF) models perform comparable to each other in accuracy, true negative, and true positive rates. Despite the fact that the models in the second and third rows were selected based on balancing the training true negative and

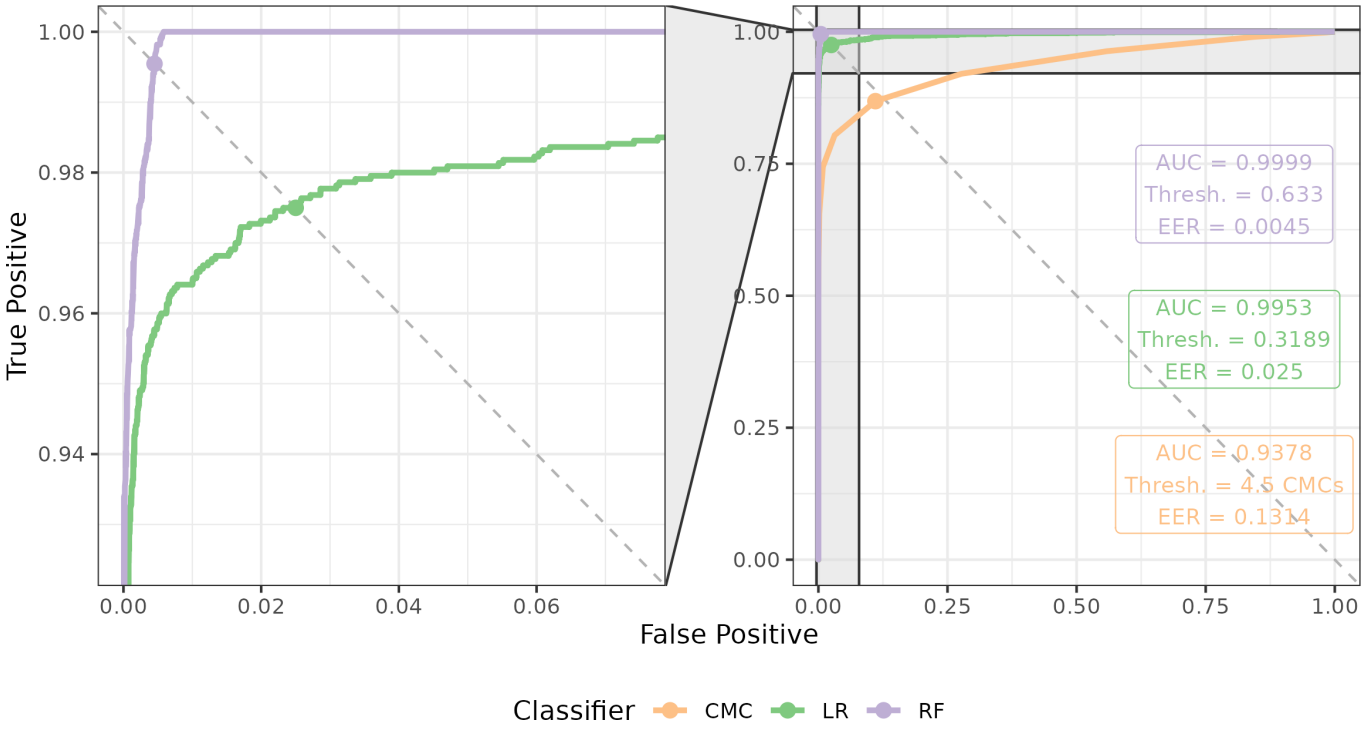


FIG. 5. ROC curves for logistic regression (LR), random forest (RF), and Congruent Matching Cells classifiers. On the left, we zoom into the top-left corner of the ROC curve plot to better distinguish between the LR and RF curves. We see that the two statistical model classifiers have higher area under the curve (AUC) and lower equal error rate (EER) values than the CMC method. We also show the score classification cutoffs (Thresh.) used for each of the four models to achieve the equal error rate values.

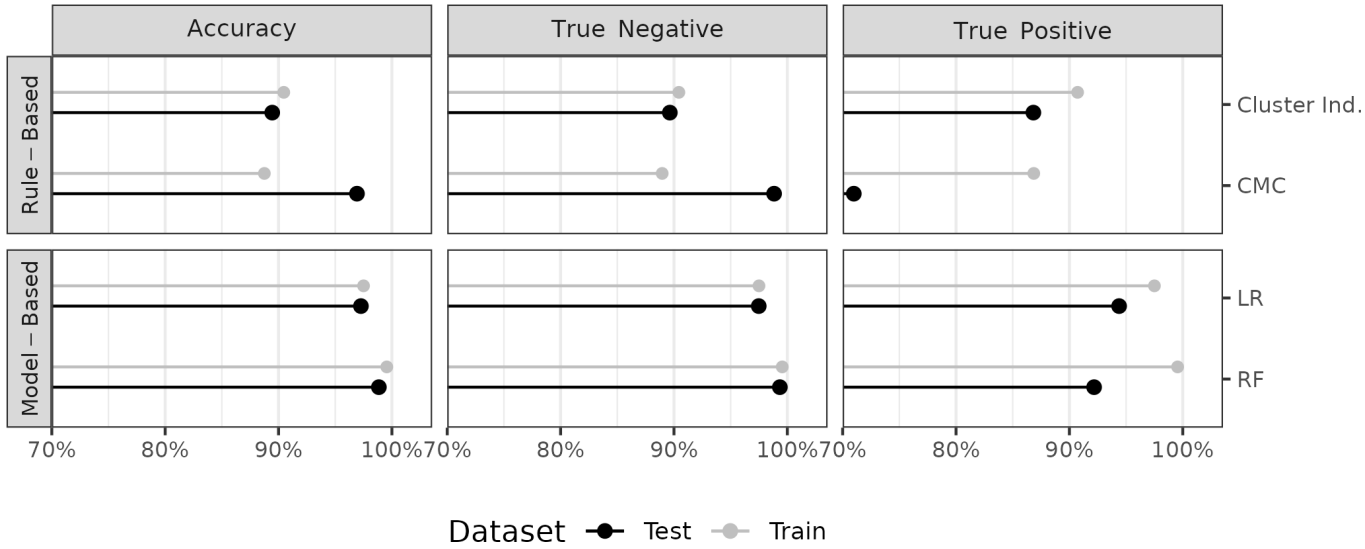


FIG. 6. We summarize classification accuracy, true negative, and true positive rates for both the training and testing results, represented as gray and black points/lines respectively, for four binary classifiers. Our primary interest is the test data results, but visualizing the training data results allows us to assess the generalizability of the classifiers after training. The first row summarizes results from a decision rule classifier based on whether the DBSCAN algorithm identifies a cluster among a comparison's cell-wise registrations. The second row summarizes results from a classifier based on the Congruent Matching Cells method. The third and fourth rows show results from training/testing classifiers based on a random forest (RF) and logistic regression (LR).

true positive rates, we note that these rates for the test data are not as well-balanced; namely, the true negative rates still tend to be larger than the true positive rates. Below, we explore this discrepancy by analyzing the contribution of various test firearms towards the true positive rates.

The third row of Figure 6 summarizes the classification results based on using all 19 ACES features. If we compare the "$C_0$ + Registration"-trained models in the second vs. the "All ACES"-trained models in the third row, we see that the addition of the other ACES features leads to improved test true negative and true positive rates (and consequently overall accuracy) with the most noticeable gains observed in the true positive rate. Across all five models, the All ACES-trained logistic regression model has the largest overall test accuracy and true positive rates of 97.68% and 95.94%, respectively. The All ACES-trained random forest model has the largest overall true negative rate of 97.87%, although the All ACES, logistic regression model is a close second at 97.81% (see Table V for more details).

### C. Similarity Score Investigation

While it's useful to consider the accuracy, true negative, and true positive rates to compare various models, forensic examiners would likely not use the binary classification returned by a model in casework. Instead, they would consider the match probability predicted by the model as a similarity score and incorporate it into their decision-making process. As such, we also consider the distribution of the predicted similarity scores for matching and non-matching comparisons. Figure 7 shows a dot plot of the predicted similarity scores for the 41,769 non-match and 3,181 match comparisons in the test set. Specifically, these probabilities are predicted by the logistic regression model selected to maximize the AUC based on the full ACES feature set. As we expect, few non-match comparisons have large similarity scores, which justifies the low false positive rate observed in Figure 6. However, there is a surprising number of matching comparisons that also have a low match probability.

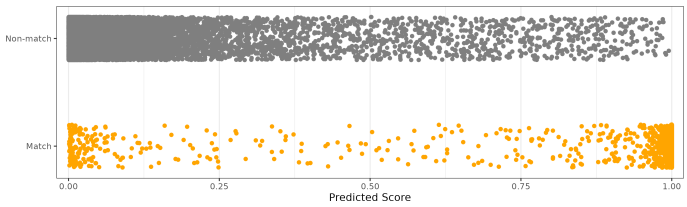


FIG. 7. A dot plot of the predicted similarity scores for the non-match and match comparisons in the test set based on a logistic regression model. As we expect, the non-match comparisons tend to have a low match probability. However, we see that there are many matching comparisons that also have a low match probability.

To explain the matching comparisons with low similarity scores, we visualize in Figure 8 the predicted similarity scores for matching test comparisons distinguished by the 15 test firearm ID. We see that the firearm T has far more matching comparisons with low similarity scores compared to the other 14 test firearms. This is further underscored by the right side of the Figure 8, which shows the ratio of misclassifications to total comparisons for every pair of test firearms based on the same logistic regression model used in Figure 7. The main diagonal shows the false negative misclassifications while the off-diagonal shows the false positives. We use blank tiles for comparisons where 0 misclassifications occurred. We see that the false negative rate for firearm T of 27.1% is far greater than that of other firearm pairs. The 95 false negative firearm T comparisons comprise 76% of all 125 false negative test comparisons and about 3% of all 3,181 matching test comparisons. In sum, the model performs distinctly worse at identifying matching comparisons from firearm T compared to the other firearms, which partially explains the lower test true positive rates noted in Figure 6. Upon visual inspection of the scans from firearm T, we noted a lack of consistent markings on their surfaces, which isn't the case for scans from other test firearms.

### D. Feature Importance

Finally, we consider the relative importance of the 19 ACES features by fitting 10 replicate random forests using the full ACES feature set with fixed random seeds. For each replicate, we measure a variable's importance using the Gini Index, which measures the probability of making a misclassification for a given model (Hastie *et al.*, 2001). A larger decrease in the Gini Index corresponds with higher importance. Figure 9 shows the distribution of the mean Gini Index decrease for the 19 ACES features. Noting the log scale on which these points are plotted, we see that the most important features consist of a combination of density-based features $C_0$, $C$, and $\Delta_{\text{trans}}$ and registration-based correlation features $\overline{\text{cor}}_{\text{cell}}$ and $\text{cor}_{\text{full}}$. In general, the visual diagnostic features tend to have lower importance scores compared the registration and density-based features. We discuss the sensitivity of these importance scores to various algorithm parameter choices in the next section.

## IV. DISCUSSION

### A. Comparison to CMC Methodology

We use a $C_0$-based classifier as a baseline because it is analogous to the classification rule proposed in Zhang *et al.* (2021). Similarly, the cell-based registration features are based on the the same cell-based comparison procedure used in Song (2013) and summarized in cell-based comparison. Together, we consider $C_0$ and the registration-based features a fusion of previously proposed cartridge case similarity scoring algorithms. This

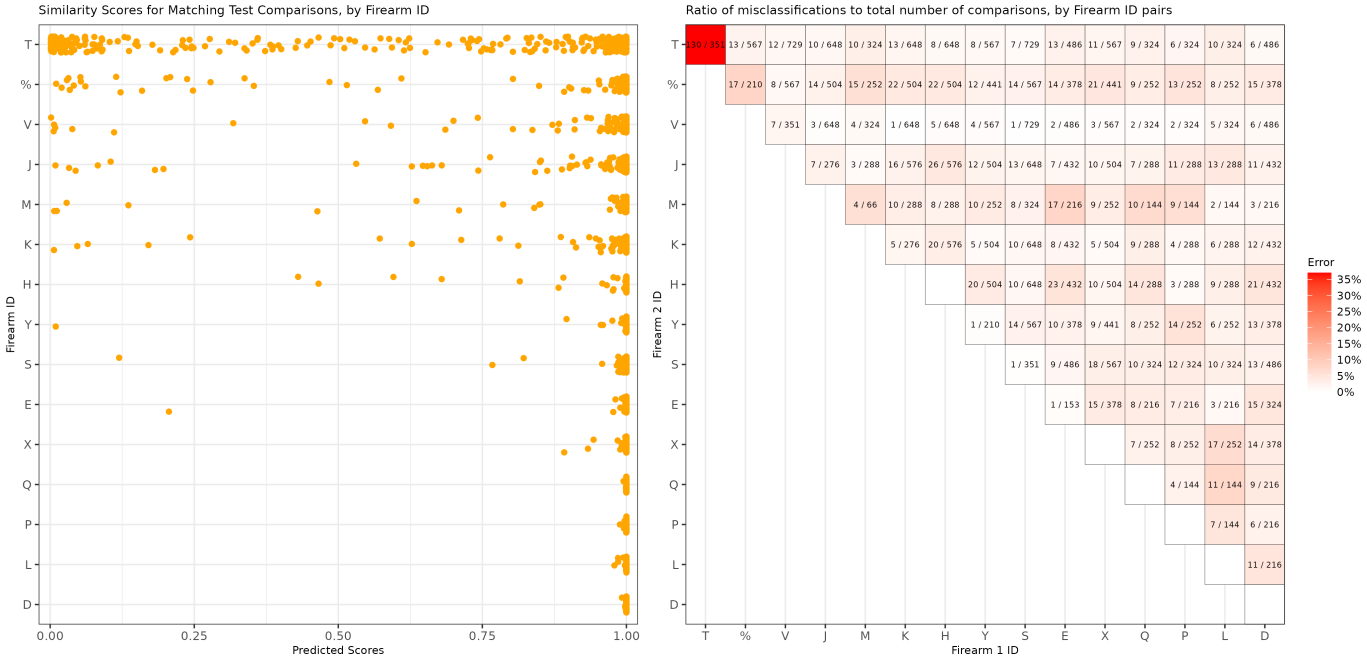


FIG. 8. (Left) A dot plot of the predicted similarity scores for the match comparisons in the test set based on a logistic regression model, separated by firearm. We see that firearm T has more matching comparisons with low similarity scores than the other test firearms. (Right) Misclassifications divided by total number of pairwise comparisons for each pair of test firearms based on the same logistic regression model. We do not show comparisons with 0 misclassifications. We note that the proportion of misclassified matching comparisons from firearm T of 27.1% is much higher than that of other comparisons.

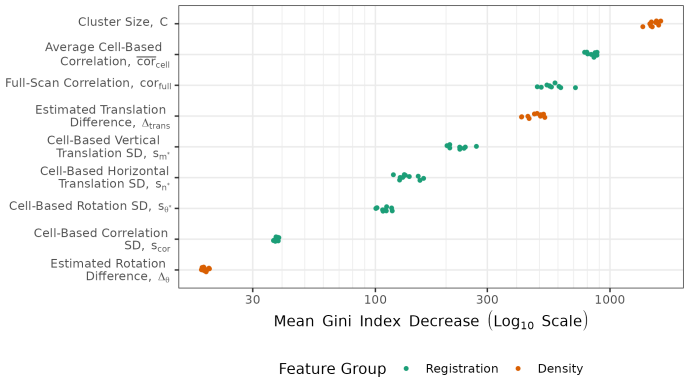


FIG. 9. Variable importance measures from fitting a random forest to the training data set, repeated 10 times under various random seeds. The top features consist of density-based features $C$ and $\Delta_{\text{trans}}$ and registration-based features $\overline{\text{cor}}_{\text{cell}}$ and $\text{cor}_{\text{full}}$. We plot points on a log scale and vertically jitter them for visibility.

is why we fit separate classifiers based on these features for the training and testing results shown in Figure 8. Table II summarizes the similarities between the ACES algorithm and the algorithms proposed in Zhang *et al.* (2021) and Song (2013). Another key difference between ACES and both of the previous algorithms is the training/testing procedure used to optimize and validate model parameters.

Table III shows the test classification error rates of the Congruent Matching Cells (CMC) algorithm proposed in Song (2013), the $C_0$-based classifier like the one proposed in Zhang *et al.* (2021), and the two ACES logistic regression models selected to balance the true negative and true positive rates and maximize the classification accuracy. We obtained the CMC results by applying the implementation available in the `cmcR` R package (Zemmels *et al.*, 2022) on the same test data set used in the Results section. We used the optimization procedure described in Zemmels *et al.* (2023) to select CMC parameters. The $C_0$-based error rates are the same as those shown in the first row of Figure 6. The ACES logistic regression models perform better than the other classifiers on this test data set, most notably when compared to the CMC method in identifying matching cartridge case pairs. Interestingly, the $C_0$-based classifier has a lower false negative error rate compared to the All ACES-trained, maximum accuracy logistic regression model, although it has a much higher false positive rate.

Both the registration and density-based features aim to measure similarities between two cartridge case surfaces. These features embody the notion assumed in the CMC methodology that matching cartridge cases should have similar markings, so their cell-based correlations should be large and estimated registrations should agree. However, Figure 3 demonstrates that even non-matching

| Original Paper | Similarities to ACES | Original Use | ACES Use |
|---|---|---|---|
| Song (2013) | Use algorithm 2 to estimate cell-based registrations | Call cells Congruent Matching Cells if their registrations are close to a reference value. Classify a cartridge case pair as a match if the CMC count is at least 6. | Compute six summative features based on full-scan and cell registrations. Use features in a classifier model. |
| Zhang *et al.* (2021) | Use DBSCAN algorithm to identify cells that reach a consensus registration | Classify a cartridge case pair as a match if a DBSCAN cluster is identified. | Compute four numerical features based on DBSCAN clusters across both comparison directions. Use features in a classifier model. |

TABLE II. Comparison of the ACES algorithm to previous work. Although ACES shares similarities to previously-proposed algorithms, it includes additional nuance by computing features across both comparison directions and using these features in a classifier model.

| Classification Method | Error (%) | False Positive (%) | False Negative (%) |
|---|---|---|---|
| CMC method | 3.9 | 2.3 | 25.8 |
| Only $C_0$ feature | 9.5 | 9.6 | 9.2 |
| ACES LR | 2.3 | 2.2 | 3.8 |

TABLE III. Testing classification error, false positive, and false negative rates for four types of classifier models. The CMC method results are derived from the implementation available in the `cmcR` R package Zemmels *et al.* (2022). The "Only $C_0$ feature" classifier is analogous to the classification rule used in Zhang *et al.* (2021). The last row shows results from the Logistic Regression classifier trained on the all 19 ACES features.

cartridge case pairs may share similar markings. We are bound to find similarities if that is all we look for, so it is worthwhile to also consider dissimilarities. The visual diagnostic features accomplish this by partitioning scans into similar and different regions. The similarities vs. differences ratio and labeled neighborhood size features measure how extreme the differences are between two scans while the differences correlation features determine whether there are similarities among the different regions.

This direct comparison of the surface values aligns with the Theory of Identification which says that an examination should involve the comparison of the "relative height or depth, width, curvature and spatial relationship" of cartridge case impressions (AFTE Criteria for Identification Committee, 1992). Comparison algorithms like ACES will inevitably be used to augment the opinion of a forensic examiner, who may need to present algorithmic results to judges or juries as part of their expert testimony. As such, it is important that forensic examiners are able to interpret and explain the results of a comparison algorithm. The visual diagnostic features are useful for explaining the behavior of the algorithm in a manner that aligns with more traditional identification theory.

## B. Model Selection Considerations

Our intention in fitting the logistic regression and random forest classification models using different feature sets is to explore each model's strengths and weaknesses. A critical step in putting the ACES algorithm into practice will be to settle on a single model. Pragmatically, it seems reasonable to choose the model with the highest estimated accuracy on available test data. However, we noted that models trained by this optimization criterion on imbalanced data tend to over-classify the majority class. This is the case for the CMC method results we summarized in Table III, but is also true for ACES statistical models trained to maximize overall accuracy. For example, if we were to shift the similarity score classification threshold for the All ACES logistic regression model to maximize the overall accuracy on the training data, the resulting score threshold is 0.54 with test accuracy, true negative, and true positive rates of 99.4%, 99.9%, and 92.4%. Given the large true negative rate, we might favor this model from an ethical perspective since misclassifying a truly non-matching cartridge case pair may incriminate an innocent individual. However, the true positive rate is considerably lower than the "balanced" results summarized in Figure 6. Further exploration of different optimization criteria is warranted.

Another aspect to consider when choosing a model is interpretability and explainability. If an algorithm is applied in forensic casework, then evidentiary conclusions derived from the algorithm's output will inevitably be presented to a non-expert judge or jury. More interpretable models are easier to understand, and therefore should be preferred. The classification behavior of the logistic regression and classification tree models are arguably easier to explain than the random forest model. For example, the logistic regression model parameters can be understood in terms of the estimated increase in odds of a match. Paired with its comparable performance to the random forest, we propose the logistic regression

model with all 19 ACES features as the preferred model that balances interpretability with accuracy.

## V. CONCLUSION

## ACKNOWLEDGMENTS

This work was partially funded by the Center for Statistics and Applications in Forensic Evidence (CSAFE) through Cooperative Agreement 70NANB20H019 between NIST and Iowa State University, which includes activities carried out at Carnegie Mellon University, Duke University, University of California Irvine, University of Virginia, West Virginia University, University of Pennsylvania, Swarthmore College and University of Nebraska, Lincoln.

We would like to thank the technicians and staff at the Roy J. Carver High Resolution Microscopy Facility for collecting the topographical scans used in this paper.

## Appendix A TRAINING FEATURE DISTRIBUTIONS

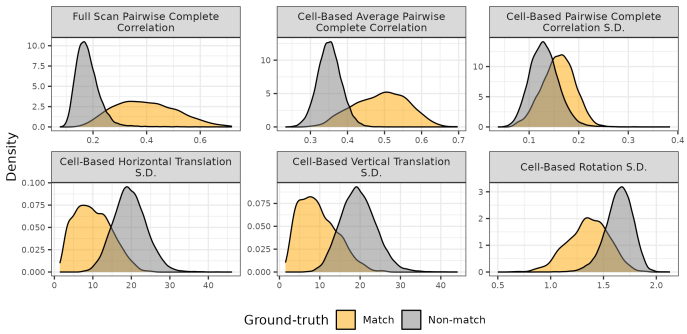


FIG. 10. Density plots of the Registration-Based features for 21,945 cartridge case pairs. The standard deviation of the cell-based registrations distinguish between match and non-match pairs better than the mean values.

## Appendix B MODEL RESULTS

| Classifier | Accuracy | True Negative | True Positive |
|---|---|---|---|
| Cluster Ind. | 90.5% | 90.4% | 90.7% |
| CMC | 88.6% | 89.0% | 86.9% |
| LR | 97.5% | 97.5% | 97.5% |
| RF | **99.6%** | **99.6%** | **99.6%** |

TABLE IV. Training accuracy, true positive, and true negative rates for the 4 binary classifiers. This table shows a numeric summary of the results shown in the Results section. We bold the largest values in each column for emphasis.

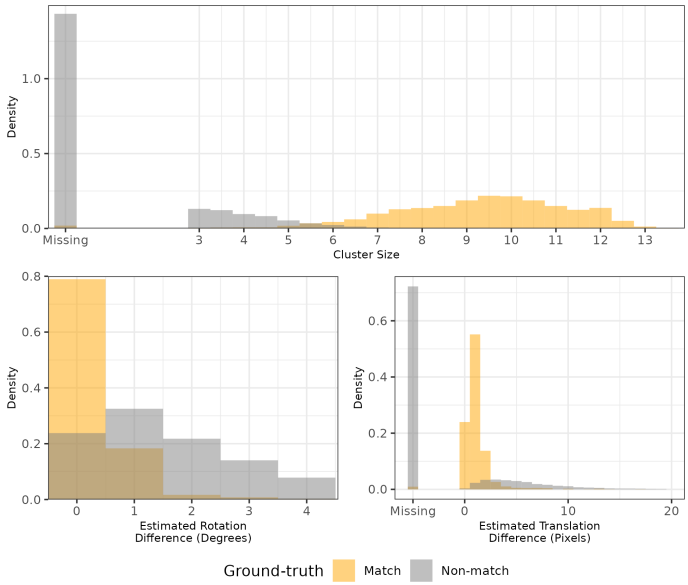


FIG. 11. Distributions of the density-based features for 21,945 cartridge case pairs. The Cluster Size and Estimated Translation Difference features may be missing (`NA`) if no DBSCAN cluster is identified, which commonly occurs for non-matching cartridge case pairs as evidenced by the stacked bar chart in the top left.

| Classifier | Accuracy | True Negative | True Positive |
|---|---|---|---|
| Cluster Ind. | 89.4% | 89.6% | 86.8% |
| CMC | 96.9% | 98.8% | 71.0% |
| LR | 97.3% | 97.5% | **94.4%** |
| RF | **98.9%** | **99.3%** | 92.2% |

TABLE V. Test accuracy, true positive, and true negative rates for the 4 binary classifiers. This table shows a numeric summary of the results shown in the Results section. We bold the largest values in each column for emphasis.

## Appendix C CONGRUENT MATCHING CELLS ALGORITHM CRITERIA

This section will provide a more thorough definition of the classification criteria used in the Congruent Matching Cells algorithm as proposed in Song (2013). This section assumes that the reader is familiar with the notation and algorithms, algorithm 1 and algorithm 2, introduced in II.

[1]The AFTE range of conclusions also permits the examiner to decide that the evidence is *unsuitable* for examination, which can occur if evidence quality is poor; for example, a fragment of a cartridge case is recovered rather than a full cartridge case.

[2]This assumption of equally-sized, square matrices is easily enforced by padding the matrices with additional missing values. Due to the presence of (structurally) missing values around the breech

face impression region, additional padding does not interfere with the structure of the scan.

[3]In reality, the true aligning registrations in the two comparison directions are opposites of each other. However, because we compare discretely-indexed arrays using a nearest-neighbor interpolation scheme, the estimated registrations may differ slightly.

[4]If more than one cluster is identified, we binarize the points based on whether they were assigned to any cluster or if they are a noise point and proceed as if there is only one cluster. We assume that two or more clusters form only because of the course rotation grid considered. Were a finer grid used, the points would coalesce into a single cluster around the true translation value. This assumption has empirical support through our experimentation.


AFTE Criteria for Identification Committee (**1992**). "Theory of identification, range striae comparison reports and modified glossary definitions," AFTE Journal **24**(3), 336–340.

Baldwin, D. P., Bajic, S. J., Morris, M., and Zamzow, D. (**2014**). "A study of false-positive and false-negative error rates in cartridge case comparisons," Technical Report (Defense Technical Information Center), https://doi.org/10.21236/ada611807.

Breiman, L. (**2001**). "Random Forests," Machine Learning **45**(1), 5–32, http://dx.doi.org/10.1023/A:1010933404324, doi: 10.1023/a:1010933404324.

Brown, L. G. (**1992**). "A survey of image registration techniques," ACM Computing Surveys **24**(4), 325–376, https://doi.org/10.1145/146370.146374.

Chen, Z., Song, J., Chu, W., Soons, J. A., and Zhao, X. (**2017**). "A convergence algorithm for correlation of breech face images based on the congruent matching cells (CMC) method," Forensic Science International **280**, 213–223, https://doi.org/10.1016/j.forsciint.2017.08.033.

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (**1996**). "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, AAAI Press, p. 226–231, doi.org/10.5555/3001460.3001507.

Fernández, A., García, S., Galar, M., Prati, R. C., Krawczyk, B., and Herrera, F. (**2018**). *Learning from Imbalanced Data Sets* (Springer International Publishing), https://doi.org/10.1007/978-3-319-98074-4.

Hastie, T., Tibshirani, R., and Friedman, J. (**2001**). Springer Series in Statistics *The Elements of Statistical Learning* (Springer New York Inc., New York, NY, USA).

James, G., Witten, D., Hastie, T., and Tibshirani, R. (**2013**). *An Introduction to Statistical Learning: with Applications in R* (Springer), https://faculty.marshall.usc.edu/gareth-james/ISL/.

Kuhn, M. (**2022**). *caret: Classification and Regression Training*, https://CRAN.R-project.org/package=caret, r package version 6.0-91.

Liaw, A., and Wiener, M. (**2002**). "Classification and regression by randomforest," R News **2**(3), 18–22, https://CRAN.R-project.org/doc/Rnews/.

MacQueen, J. B. (**1967**). "Some methods for classification and analysis of multivariate observations," in *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, edited by L. M. L. Cam and J. Neyman, University of California Press, Vol. 1, pp. 281–297.

National Research Council (**2009**). *Strengthening Forensic Science in the United States: A Path Forward* (The National Academies Press, Washington, DC), https://doi.org/10.17226/12589.

President's Council of Advisors on Science & Technology (**2016**). "Forensic science in criminal courts: Ensuring scientific validity of feature-comparison methods," https://obamawhitehouse.archives.gov/sites/default/files/microsites/ostp/PCAST/pcast_forensic_science_report_final.pdf.

R Core Team (**2017**). *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, https://www.R-project.org/.

Song, J. (**2013**). "Proposed "NIST Ballistics Identification System (NBIS)" Based on 3D Topography Measurements on Correlation Cells," American Firearm and Tool Mark Examiners Journal **45**(2), 11, https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=910868.

Thompson, R. (**2017**). *Firearm Identification in the Forensic Science Laboratory* (National District Attorneys Association), p. 32, https://doi.org/10.13140/RG.2.2.16250.59846.

Tong, M., Song, J., and Chu, W. (**2015**). "An Improved Algorithm of Congruent Matching Cells (CMC) Method for Firearm Evidence Identifications," Journal of Research of the National Institute of Standards and Technology **120**, 102, https://doi.org/10.6028/jres.120.008.

Venables, W. N., and Ripley, B. D. (**2002**). *Modern Applied Statistics with S*, fourth ed. (Springer, New York), http://www.stats.ox.ac.uk/pub/MASS4, iSBN 0-387-95457-0.

Vorburger, T. V., Yen, J. H., Bachrach, B., Renegar, T. B., Filliben, J. J., Ma, L., Rhee, H. G., Zheng, A., Song, J. F., Riley, M., Foreman, C. D., and Ballou, S. M. (**2007**). "Surface topography analysis for a feasibility assessment of a national ballistics imaging database," Technical Report NIST IR 7362, https://nvlpubs.nist.gov/nistpubs/Legacy/IR/nistir7362.pdf, doi: 10.6028/NIST.IR.7362, edition: 0.

Zemmels, J., Hofmann, H., and VanderPlas, S. (**2022**). *cmcR: An Implementation of the 'Congruent Matching Cells' Method*, r package version 0.1.9.

Zemmels, J., VanderPlas, S., and Hofmann, H. (**2023**). "A study in reproducibility: The congruent matching cells algorithm and cmcR package," The R Journal **14**(4), 79–102, https://doi.org/10.32614/rj-2023-014, doi: 10.32614/rj-2023-014.

Zhang, H., Zhu, J., Hong, R., Wang, H., Sun, F., and Malik, A. (**2021**). "Convergence-improved congruent matching cells (CMC) method for firing pin impression comparison," Journal of Forensic Sciences **66**(2), 571–582, https://onlinelibrary.wiley.com/doi/abs/10.1111/1556-4029.14634.