The author(s) shown below used Federal funding provided by the U.S. Department of Justice to prepare the following resource:

Document Title:	**Advanced Scan Matching, Scalable Search, and Visualization Tools for the Analysis of 3D Scans of Cartridge Casings in Firearm Forensics**
Author(s):	**Ryan Lilien**
Document Number:	**252774**
Date Received:	**March 2019**
Award Number:	**2016-DN-BX-0182**

Final Research Performance Progress Report - Cover Page

Federal Agency and Organization Element: Department of Justice, Office of Justice Programs

Federal Grant or Other Identifying Number: 2016-DN-BX-0182

Project Title: Advanced Scan Matching, Scalable Search, and Visualization Tools for the Analysis of 3D Scans of Cartridge Casings in Firearm Forensics

PD/PI Name, Title, Contact Info: Ryan Lilien, Chief Scientific Officer, Cadre Research Labs; 420 W Huron St, Suite 204; Chicago, IL 60654

Name of Submitting Official: Ryan Lilien

Submission Date: July 25, 2018

DUNS: EIN:

Recipient Organization: Cadre Research Labs, LLC (small business)

Recipient Identifying Number (if any): N/A

Project/Grant Period: Start: 1/1/2017, End: 4/30/2018

Reporting Period End Date: 4/30/2018

Report Term or Frequency: Twice a year

Signature of Submitting Official:

1 Project Purpose and Background

Forensic science will continue to benefit from advances in 3D imaging and advanced algorithms for image and statistical analysis. Several reports, including two from the National Academy of Sciences, have called for additional research in this area [1, 2, 3]. Development of new instrumentation and comparison algorithms can help address the identified challenges. These novel methods can increase the quality and reduce the cost of forensic analyses. They will not replace the firearms examiner; rather, new tools will allow the human expert to better and more efficiently complete their analysis. For example, accurate 3D measurements can be shared between labs, can serve as the basis for blind verification, and act as the raw data for objective comparison algorithms. In 2013, we began development of a 3D surface topography imaging and analysis system for firearm forensics. Several next steps will extend our work and enable continued adoption of 3D in the crime lab. This proposal contains three important aims. First, we improved the matching algorithm to improve recall for minimally-marked cartridge cases. Second, we significantly increased the speed of database search by developing a distributed computing version of our search method. Finally, we created and released free 3D viewer software for the firearm and toolmark examination community. The completed work includes critical enabling steps towards the adoption of 3D scanning technology in the forensic lab.

The research work was completed by Cadre Research Labs, a scientific computing contract research organization, working in collaboration with GelSight Inc, a company formed by the MIT researchers who developed the GelSight surface topography imaging technology. The two companies collaborate closely with Todd Weller, a firearms identification specialist and Criminalist with former affiliation with the Oakland Police Department and who is now in private practice. We continue to collaborate with colleagues at NIST as well as federal, state, and local crime labs. These collaborators continue to be excellent partners and provide both scans and constructive feedback.

2 Project Design

The one year project included three aims which continued the R&D of our novel technology so that it best suits the needs of the forensic community. All three aims were focused on improving usability of 3D technology in the crime lab. The ability to improve database recall for minimally-marked cartridge cases while not sacrificing match accuracy (Aim 1) can greatly enhance an examiner's ability to make connections between criminal cases and provide objective verification for firearms examinations. In Aim 1 we present a new scoring function to complement (not replace) our current function. As databases continue to expand, rapid database search will become increasingly important. In Aim 2 we developed a distributed computing version of our search algorithm. The new approach is capable of running on a single machine (as it does now) or on a large compute cluster (where it obtains a speed-up proportional to the number of worker nodes recruited). Finally, in Aim 3 we developed and supported free 3D viewing software for surface topographies. The availability of a free viewer enables sharing between labs, even if a lab does not have a 3D scanner. All three aims benefit from increased adoption of the X3P file format. All 3 proposed aims were successfully completed during the project period.

3 Materials and Methods

Each of the three aims represent independent work and will be discussed separately. Methods have been abbreviated to conform to page limits. In this section we describe the general approach for each aim. In the Results and Analysis section we describe the experimental performance and details of each aim.

3.1 Matching Algorithm for Minimally-Marked Cartridge Cases (Aim 1)

Cartridge case matching algorithms, such as ours [8], can achieve strong performance when comparing well-marked cartridge cases (Figure 1). However, these methods may miss a known-match if the cartridge cases are minimally-marked (Figure 2). The reason for this is understandable. Well-marked cartridge cases have significant surface geometry for matching whereas minimally-marked cartridge cases have few surface features. Minimally-marked cartridge cases often have large areas of the surface appearing completely devoid of features. An algorithm capable of identifying correct matches while minimizing the false-positive rate (*i.e.*, the number of known-non-matches scoring above the significance threshold) would be extremely useful for the forensic community. In this aim we developed a version of our scoring function for minimally-marked cartridge cases which focuses only on recall. Recall measures the percentage of time where a correct known match is ranked at the top (number one) position when a reference surface is compared against a database of topographies.

3.1.1 Data Set Design

We assembled a set of cartridge cases including three test-fires from each of 210 firearms from 32 manufacturers[1]. Ammunition was either Remington or Winchester. Primers were either brass (46, 22%) or nickel (164, 78%). The set included 68 (32%) 0.40 S&W, 43 (20%) 0.45 ACP, and 99 (47%) 9mm Luger. Two of the three test-fires from each firearm were randomly selected to represent 210 pairs of knowns (420 cartridge cases) with the remaining test fire for each firearm composing the set of 210 unknowns. The two knowns were referred to as cartridge cases A and B. The third cartridge case is considered the unknown cartridge case C. To identify a set of minimally-marking firearms we compared the known test fires (A and B) to each other for each firearm using only the breech-face impression and our feature-based comparison algorithm. Note that neither the aperture shear nor firing-pin impression were used in this test. This decision was made because we wanted to focus on the ability to improve recall using the breech-face impression only. If the AB comparison score was below a predefined threshold, then we considered the firearm to be minimally-marking. Using this criteria 122 (58%) of the firearms were sufficiently well marking and 88 (42%) of the firearms were considered minimally-marking. The minimally-marking firearms were relatively evenly split among the calibers with 45% of the 0.40 S&W being minimally-marked, 33% of the 0.45 ACP being minimally-marked, and 43% of the 9mm Lugers being minimally-marked. Primer material was also evenly split with 41% of the brass primers and 42% of the nickel primers being considered minimally-marked. For the tests below we compared all cartridge cases to each other regardless of caliber to increase the effective difficulty and size of our test set. In practice, one would only compare to cartridge cases of the same or compatible calibers.

Examples of cartridge cases falling into each category are shown in Figures 1 and 2. Figure 1 shows examples of extremely well marked (top) and slightly less well marked (bottom) cartridge cases that are correctly identified by our scoring algorithm. These firearms are part of the 122 firearms listed above with sufficient markings. Figure 2 shows examples of cartridge cases from firearms in the set of 88 that are labeled minimally-marked. Figure 2-Top shows two cartridge cases that are minimally-marked but which are indeed recalled using the new algorithms described below. Figure 2-Bottom shows two cartridge cases that are minimally-marked and for which recall fails even with the new algorithms. As can be seen, these cartridge cases may have insufficient breech-face impression marks for identification by any human or machine expert.

[1] Firearm makes included: AA Arms, Beretta, Browning, Bryco Arms, Citadel, Colt, Essex Arms, European American Armory, FEG, Glock, H&K, Haskell, Hi-Point, Intratec, Kahr, Kel-Tec, Kimber, Kriss Vector, Llama Gabilondo, Norinco, Para Ordnance, Randall, Rock Island, Ruger, SCCY, Sig Sauer, S&W, Springfield Armory, Star Bonifacio, Stoeger, Taurus, Walther

3.1.2 Algorithm Design

Because our current scoring function performs well when comparing most cartridge-cases, we created a second, separate scoring function for use with minimally-marked cartridge cases. Towards this development we developed and evaluated a number of algorithmic variants each designed to improve recall. There are two parts of each algorithmic variant. First, the algorithm must analyze a surface to identify surface-features. Second, the algorithm must compare the detected features to quantify the degree of geometric similarity. The combinations of the following variants of feature detection and feature matching were considered in our experiments.

- **Increased Numbers of Features**: Surface features consist of geometric structures present on the scan surface. Most feature detection approaches utilize a local quantified measure of surface roughness or texture to identify regions of the surface with sufficient detail to be considered a feature. A numeric threshold is used to identify which features to keep and which to discard. If the threshold is set too low then there's a risk of non-informative features being included. If the threshold is set too high then there's a risk that some informative features may be ignored. Our current scoring function utilizes a threshold designed to avoid false-positive matches. For the identification of minimally-marked cartridge cases it may be useful to lower the threshold and include additional features. In completing Aim 1, we considered both the original threshold as well as a lowered threshold for feature detection.

- **Examiner-Guided Alignments**: In the examiner-guided approach the software prompts the user to annotate the regions of the reference scan that they believe are the most important and informative for identifying a match. The computer then detects surface features (similar to those in our current approach) that are located in the marked region. When evaluating each candidate scan, the algorithm looks for these 'important' features to be arranged in the same geometric positioning as they appear in the reference. While locating these marked features the remainder of the surface is ignored.

- **Self-Consistent**: The self-consistent approach is similar to the examiner-guided approach; however, rather than prompting the user to annotate the scan surface it is done automatically. The self-consistent approach requires at least two known test fires from the same firearm. These two (or more) known test fires are compared to each other. Any regions with consistently identified geometric features are marked as the 'important' features on which to match. As with the examiner-guided approach, a score is computed based on the total number of marked features identified between a reference and candidate. This corresponds to the set-theory operation of taking the intersection of the feature sets of A and B.

- **Meta-Surface**: The meta-surface approach extracts features from several known test fires and combines all features into a single dense meta-surface. The approach requires at least two known test fires from the same reference firearm. The reference test fires are compared and aligned to each other. All extracted features are combined (*e.g.*, merged) into a single meta-surface. The meta-surface is used as a new reference and is compared against the candidate surface. This corresponds to the set-theory operation of taking the union of the feature sets of A and B.

3.2 Scalable Computing for Remote Search (Aim 2)

The speed required to search a database is a function of the number of scans in the database and the time required to compare each individual scan. As databases increase in size, so do search times. In Aim 2 we will expand our search algorithm to a scalable compute environment. This will allow both extremely fast search as well as remote data access.

Modern high-performance computing typically utilizes multiple general purpose compute nodes arranged as a compute cluster. For example, a cluster may consist of hundreds of independent compute nodes (*e.g.*, individual computers) physically arranged on large racks in a machine room. The nodes typically run the Linux operating system and are configured in a Master-Worker arrangement (Fig 3). When software is run on multiple machines is it called a distributed computing task. For example, consider the job of resizing one million large images. The master node doles out tasks to each worker node. When a worker node completes a task, the result is returned to the master and the master assigns the worker a new task. Task management is facilitated by a queue which is an ordered list of jobs to complete. In our example, the master node launches a number of worker nodes and places all images to be resized into a job queue. Each worker grabs an image to resize from the queue. When a worker completes the resizing operation it sends the rescaled image back to the master and obtains a new image to resize from the job queue. This process continues until all images have been rescaled. Running on k worker nodes ideally reduces the runtime by a factor of approximately k (*e.g.*, for one hundred worker nodes ($k = 100$), a task that would take 200 hours on a single machine would take only two hours)[2]. In Aim 2 we developed a parallel version of our search algorithm capable of running on both a single workstation, a large

[2]In practice it's typically not possible to achieve a speed-up of k using k nodes. Typically the speedup is 60-80% of k-fold depending on the overhead of job coordination.

dedicated compute server, or a secure scalable distributed server.

We implemented a cloud-based distributed computing platform called the Cadre Nexus using the Amazon Web Services (AWS) compute resource. AWS is designed to provide scalable on-demand computing for academic, industry, and government groups. AWS consists of dozens of core resources. The primary resources used in our setup include: Cognito (for user account management), S3 - Simple Storage Service (for disk storage), RDS - Relational Database Service (for database), EC2 - Elastic Compute Cloud (for compute servers), and SQS - Simple Queue Service (for batch computing scheduling). This architecture includes components to handle user authentication, data storage, and data search. Several configuration options are specified in configuration files. At a high-level, X3P files are stored in S3 and organized in the RDS database. Our comparison algorithm sits on one or more virtual machines running on EC2 compute nodes. All interactions with the Nexus require that the user be authenticated via the Cognito service. All requests are handled by the AWS API Gateway. We setup a relational database with database tables for objects such as users, regions, cartridge cases, scan topography data (X3P files), scan meta-data (*e.g.*, cartridge case caliber), and match results.

In the first application, we setup the Nexus to store X3P data files. These files can be downloaded to user computers or viewed in a prototype web-browser based 3D topography viewer (Figure 4). The web viewer is designed to implement many features of our X3P desktop software. The user can rotate, translate, and zoom the 3D surface topography. Users can also adjust a virtual light to highlight specific features of the topographic surface.

The second application implements a distributed computing version of our comparison algorithm. The architecture implements a master-worker framework (Figure 3). When a user requests that a set of cartridge cases be compared, a series of computer programs (scripts) launch to coordinate the server activities. First, the software activates the requested number of EC2 compute nodes with each node receiving a copy of the comparison software. Second, the software sets up a queue (or list) of all required pairwise comparisons. This list is managed by the SQS Queue which is monitored by the AWS Batch service for distribution to the EC2 compute nodes (Figure 5). Each EC2 node repeatedly runs the comparison algorithm to complete a set of surface topography comparisons. Upon completing a comparison, the compute node checks the queue and begins work on computing the next required pairwise comparison. When the work queue is empty the compute job is complete. In this manner it is possible to utilize any number of EC2 compute nodes. A user (*e.g.*, our group) has a master control program which dynamically rents EC2 nodes in response to demand. For example, when the system is idle only the master control program is running. When a search request is received, the control program can recruit a number of EC2 nodes to perform the work. When the work is complete the control deactivates these nodes.

Parallel computing environments like that described above are inherently complex systems. Our setup uses more than ten different software tools, three programming languages, and much fine-coordination. This is par for the course. The good news is that the complex back-end is not exposed to the user. That is, a user wishing to interact with the Nexus does so through normal desktop software and web interfaces (for example, the web interface described above, Figure 4).

3.3 3D Viewer Software (Aim 3)

We continued to improve the overall functionality and stability of our free X3P 3D virtual viewer software. We addressed a major restriction on our earlier versions. This restriction related to the types of computer graphics cards on which our software could run. We also designed and implemented an encrypted version of the X3P file format.

4 Data Results and Analysis

In this section we summarize the experimental results of the three proposed aims. As is common with experimental algorithm design we implemented and evaluated many variants of the following methods. Space limitations require us to us describe only the best or most interesting results.

4.1 Matching Algorithm for Minimally-Marked Cartridge Cases (Aim 1)

The performance of the following methods were evaluated using ‘recall’. Recall indicates that the correct cartridge case was ranked highest among the 210 unknowns when their breech-face impression surfaces were compared using the specified method. Of the 88 minimally-marking firearms we first identified how many of them could be recalled using the current feature set and comparison methods. That is, it’s possible that although the match score is below our significance threshold that it still ranks the correct test fire highest among all candidates. A completely random (and obviously useless) comparison algorithm would have a 1 in 210 chance of guessing correctly and thus obtain a recall of 1/210 (or 0.4%). When the 88 A-cartridge cases were compared to all 210 C-cartridge cases the recall was 33/88 (38%). When the 88 B-cartridge cases were compared to all 210 C-cartridge cases the recall was 37/88 (42%). We define the average of these results to be the baseline recall (*e.g.*, 35 of 88). These results indicate that about

60% (53 of 88) of the minimally-marked cartridge cases (and 25% of the entire set of 210 firearms) are not recalled using the current algorithm. It is these firearms that we hope to recall using the proposed approaches.

The union (meta-surface) and intersection (self-consistent) methods both compare and combine the features of the two known reference surfaces (A and B). Both methods require that the two surfaces (A and B) be correctly aligned. This is inherently a challenge because these surfaces come from firearms which are minimally-marked and therefore difficult to align. We implemented a hybrid multi-stage approach to align surfaces A and B. This approach utilizes a stepwise search over 360-degrees of in-plane rotation, a cross-correlation based comparison weight, and feature-based comparisons. Overall, these approaches improve our ability to align scans A and B. As a preprocessing step to the results that follow we have automatically aligned scans A and B using this hybrid alignment method. Note that this alignment may not be correct. When the alignment is not correct some of the methods below are unlikely to succeed; however, when the alignment is correct, the proposed methods may be successful.

Meta-Surface (Union): After aligning the known test-fires for all 88 minimally-marked cartridge cases we formed the union of all features on each of scans A and B. We remove one copy of the redundant features so that only one copy remains in the union and the features are not double counted. The meta-surface algorithm performance recalls 38 of the 88 firearms. This is only slightly better than A alone (33) and B alone (37). It indicates that there is a small amount of additional information gained in combining the two scans.

Self-Consistent (Intersection): After aligning the known test-fires for all 88 minimally-marked cartridge cases we formed the intersection of all features of scans A and B. All features not common to A and B are discarded and we are only left with the features common to both A and B. This approach performed poorly as only 8 of the 88 firearms were recalled. We propose that these results are due to two phenomenon. First, because these are minimally-marked cartridge cases the number of features in the intersection is inherently very low. Therefore we were starting the search at an initial disadvantage. Second, because these are minimally-marked cartridge cases they are difficult to align perfectly which leads to a lower than expected intersected feature count. With a small number of features in the intersection there is little separation between the known matches and known non-matches.

Increased Numbers of Features: This approach increases the number of detected features for each of A and B by lowering the threshold required for significance. The thinking is that although these more subtle features are less likely to be consistently reproduced from test-fire to test-fire, the new feature set may still contain a small number of 'new' consistent and informative features not included in the original list. The tradeoff is that while increasing the number of features should increase the number of matching features for the known matches that it might also increase the random-chance matching of these subtle features in the known non-matches. We lowered the significance threshold to increase the number of detected features by approximately 50% as compared to the normal approach. Increasing the number of features only had a slight effect on recall. When scan A was compared against all 210 unknowns, the recall increased from 33 with regular features to 36 (out of 88) with increased features. The scan B recall was 37 with both the regular and expanded features.

Meta-Surface (Union) with Increased Features: We combined the two referenced ideas above to produce a unioned surface using the increased number of features for both scans A and B. The recall increased from 38 (with the regular meta-surface) to 46 (out of 88).

Examiner-Guided Alignments: In this approach a human-expert was asked to mask the surfaces of both scans A and B to indicate the regions containing informative surface geometry. Only features in the highlighted regions were used in matching. However, to simulate what would happen in a real test environment the unknown test fires (scans C) were not specially masked. For scans C, the entire breech-face impression of the unknown test-fires were used. This approach allowed the recall of two additional firearms bringing the total to 48 of 88.

A graphical summary of the results are shown in Figure 6. Figure 2-Top shows two firearms that are recalled with the above approach and Figure 2-Bottom shows two firearms that are not recalled despite the above method. Overall, the development of techniques specifically for minimally-marked cartridge cases allowed us to correctly recall 13 more firearms than the baseline (48 vs. 35) corresponding to 25% (13/(88-35)) of those firearms we could not recall with our regular methods. This raises the recall of the entire set to 81% (170 of 210). We feel this is pretty good performance on a difficult dataset given that we are only utilizing the breech-face impression mark. For example, we know that several of the non-recalled cartridge cases have strong aperture-shear marks and would likely be recalled by our aperture-shear matching algorithm. We note that it is not likely possible to achieve 100% recall as many firearms do not produce sufficient toolmarks. An informal verbal poll of examiners felt that approximately 75-85% of cartridge cases they encountered in casework could be identified by the breech-face impression marks alone. Although this suggests that our performance is pretty good we aim to continue developing these methods. Finally, we note that recall should not be confused with identification. None of the above results indicate any false positive match results. Our original feature-based matching algorithm shows no false positives on this dataset and the recall focused algorithm described here simply helps identify possible hit-candidates from a database.

4.2 Scalable Computing for Remote Search (Aim 2)

System Architecture: As described in the methods section, we developed and implemented a cloud computing platform, called the Nexus, for database storage of X3P files and high-performance computing. Databases consist of items (or records) arranged in a series of interrelated tables. Our database has records to keep track of Users, Labs, Regions, Specimens, Data Files, and Matches. Most of these records are self-explanatory with the exception of 'Specimens' which are used to keep track of individual physical items (*e.g.*, cartridge cases), 'Data Files' which are the individual X3P surface topographies or their related mask files, and 'Matches' which store the numerical match result computed between two specimens.

Users are able to connect through a prototype web interface to view items in the database (Figure 4). We are now connecting this web interface to our desktop software X3P viewer and main TopMatch software (*e.g.*, software of Aim 3).

We also developed a version of our cartridge case comparison software to run on the AWS platform. Our software was written to compile on multiple computing platforms. Our AWS architecture uses nodes with the Linux operating system and so we created a version of our comparison software capable of running under Linux. The selected algorithm is related to NIST's CMC (Congruent Matching Cells) algorithm [5, 6, 7]. Our implementation accepts as input two surface topographies (in X3P format) and a configuration file. It outputs a match score quantifying the amount of identified geometric similarity between the two scans. This cross-correlation based comparison algorithm is slower than our typical feature-based method and is a great test for a distributed computing experiment.

Each run consists of startup time and processing time. The 'startup time' includes the time required to start-up each of the worker compute notes and to setup the initial list of jobs (comparisons). Because each requested worker node starts at the same time (in parallel), the startup time is relatively constant across runs. Startup requires approximately 4 minutes regardless of the number of worker nodes and number of requested comparisons. The 'processing time' includes the time required to complete all requested comparisons. A single pairwise comparison using the implemented algorithm requires approximately 1 minute on a single worker node.

Algorithm Evaluation: We tested using a version of the Weller dataset which contained 95 cartridge cases from ten consecutively manufactured 9mm Luger firearms [9]. Scans of the primer were acquired using our TopMatch scanner at a resolution of approximately $1.5\mu m$/pixel. Data was saved as individual topography files, masked to the breech-face impression, and uploaded to the Nexus network.

Test 1 (200 Comparisons): Our first test served as a proof-of-concept and included a small subset of 200 pairwise comparisons. The first run used 1 worker node and the second run used 5 workers. Both tests had startup times of 4 minutes. The test with one worker required 3 hours and 35 minutes of processing time, corresponding to 64 seconds per comparison. The test with five workers required 47 minutes of processing time, corresponding to 14 seconds per comparison and a speed-up of 4.6x.

Test 2 (4465 Comparisons): Our second test included all 4465 unique pairwise comparisons of the 95 Weller scans. Tests were performed with 5, 10, 20, 40, and 80 worker nodes. Note that we did not test a single worker node which would have required approximately 80 hours (or 3.3 days) to complete all comparisons. Startup time for each experiment was approximately 4 minutes. A table and graphical summary of the results appears in Figure 7. The test with 5 worker nodes required a total of 19 hours 45.5 minutes, corresponding to 15.9 seconds per comparison (a 4.1x speed-up). The test with 10 worker nodes required a total of 10 hours 51.5 minutes, corresponding to 8.8 seconds per comparison (a 7.3x speed-up). The test with 20 worker nodes required a total of 5 hours 33 minutes, corresponding to 4.5 seconds per comparison (a 14.3x speed-up). The test with 40 worker nodes required a total of 2 hours 56 minutes, corresponding to 2.4 seconds per comparison (a 26.7x speed-up). The test with 80 worker nodes required a total of 1 hour 32 minutes, corresponding to 1.2 seconds per comparison (a 53.3x speed-up). Note that in parallel computing one does not get a perfect k-fold speed-up when k workers are used. This is due to the overhead required to coordinate each worker and the time required to manage / merge all the results. These results demonstrate successful design and implementation.

We note that the base implementation of our typical feature-based comparison search is approximately 20-40 times faster than the cross-correlation based method tested. The feature-based method would also benefit from the above parallel computation setup and is something we are now implementing. Using 80 compute notes, our feature-based method would require between 2.3 and 4.6 minutes to complete all 4465 comparisons (20-40x faster than 1.5hrs for 80 nodes above).

The design and implementation of the Nexus platform demonstrates a significant portion of what would be required for the next generation of 3D topography analysis tools for firearm forensics. We've demonstrated remote storage, access, and search. The use of the common X3P file format means that various 3D scan acquisition systems can both contribute to and download from the Nexus platform. The use of multiple worker nodes allows the cloud platform to scale to any number of search requests while maintaining any desired turn-around time. For example, we demonstrated that while a single computer would require 3.3 days to complete a comparison task, the same query can be completed in under 1.5 hours using a group of 80 worker nodes.

4.3 3D Viewer Software

We continued to improve our free X3P viewer software. Our software is now in use by practitioners in over eleven countries and thirty labs (academic, industry, and government). The core graphics rendering engine was rewritten to improve speed and stability. This rewrite also allows the software to run on additional and lower-end computer hardware. The software now supports NVIDIA, AMD/ATI, and Intel HD Graphics. This support means that the X3P viewer should run on the vast majority of Windows-based machines manufactured back to 2016. To further improve accessibility we've begun development of a web-based viewer (an aim not specifically part of the proposed work). The web-based viewer may be a little slower on older machines but requires no special software to install. Users will load scans from the Cadre Nexus through their web-browser.

On the other end of the spectrum, we've updated our software to run on the latest high-resolution 4k monitors. In general, older legacy software that is not 4k-ready can often run on 4k hardware; however, the increased pixel density often results in extremely small fonts and tiny unusable user interface elements. These unwanted visualization effects can make using the software extremely difficult. After our update, fonts and other user elements now scale gracefully.

Fernet uses AES with a 128-bit key for encryption resulting in a fernet token that cannot be accessed or altered without the key.

We developed an encrypted version of the X3P file format. Our implementation uses AES symmetric authenticated cryptography to ensure a file cannot be accessed or altered without the correct password. Conceptually, the encryption process can be divided into two steps. First, we derive a 256-bit encryption key from a user-supplied password. Second, we use the key to encrypt the X3P file according to the fernet format. The result of this process is an encrypted file, denoted by the 'X3PE' file extension. To create the encryption key, we used the PBKDF2-HMAC-SHA256 key derivation function. This function combines the user's password with a random 16-byte salt value and iteratively applies the HMAC-SHA256 hashing algorithm to generate a key. For this implementation, the key is base64 encoded to represent the binary data in ASCII and the salt value is stored at the beginning of an X3PE file. Once the encryption key has been created, the X3P file is encrypted. Decryption involves the same steps but in reverse. The correct password must be entered to derive the key which is first checked against the encrypted file (to ensure the password is correct) and then used to decode the data. At its core, the X3PE file is the same as a traditional X3P file with the main difference being that the entire X3P file is encrypted using a user supplied password. We've implemented both reading and writing software for the X3PE format in the Python programming language. The implementation was tested using a handful of X3P files.

4.4 Continued Deployment Study

As we have during each of our previous awards, we continue to collaborate with crime labs. Through most of the project period we had a machine setup with Thomas Fadul and his lab at the Miami-Dade crime lab in Miami-Dade county Florida. At the beginning of the project period, Ryan Lilien went down and provided a day of hands-on training to all examiners and trainees in the lab. Then during the deployment period a single trainee assumed the main point-of-contact within the lab. The Miami-Dade lab collected over 300 scans during the deployment period. Upon completion, we brought the scanner back to Chicago and upgraded it to version 3 hardware (with a batch scanning tray) and then set it up with the West Virginia State Police crime lab. After West Virginia, the machine is scheduled to go to the Michigan State Police metro Detroit lab for six months and then to the Indiana State Police for six months. Through deployments like these we continue to collect scan data, to elicit excellent feedback from practitioners, and to train examiners and trainees.

5 Scholarly Products Produced

The primary product of the proposed research is the presentation of our results and progress. At the May 2017 AFTE national meeting we gave two technical presentations. One presentation took place during the main technical session and was entitled "Validation of 3D Virtual Microscopy for Cartridge Case Comparison". This presentations were well received; we received many requests for slides after the talk. At the same meeting we co-ran a virtual microscopy workshop titled "Implementation of 3D Technology, Analysis, and Statistics for FA/TM Examinations". During the full-day workshop participants had hands-on time with our virtual microscopy software. They worked through a training tutorial and a virtual CTS test. Lilien also presented our work on validating virtual microscopy at the Miami-Dade Crime Lab (Miami, FL), the Washington DC Crime Lab (Washington DC), the Eastern Regional AFTE meeting

(FBI Organized, Fredericksburg, VA), the IPTES meeting (Arlington, VA), the NIJ Special Symposium at PITTCON (Orlando, FL), and Tulane University (New Orleans, LA). We had our journal paper titled "Development and Validation of a Virtual Examination Tool for Firearm Forensics" [4] accepted for publication in the Journal of Forensic Sciences. We aim to submit a journal version of our grant final report to the AFTE journal. The above publications and presentations continue our tradition of disseminating our research results. Over the past several years, we have presented at twenty forensic conferences and run training sessions at fourteen local, state, and federal crime labs.

6 Summary

We successfully completed the proposed aims during the project period. We developed algorithms for the examination of minimally-marked cartridge cases (Aim 1). While it is not possible to achieve the same performance on minimally-marked cartridge cases as is enjoyed with well-marked cartridge cases our work shows that it is still possible to facilitate recall with minimally-marked cartridge cases. We were able to recall approximately 25% of the minimally-marked cartridge cases in our set. Those we could not recall were extremely minimally-marked (as shown in Figure 2). In Aim 2 we developed and tested a scalable computing platform for remote search. Our solution was implemented on the AWS platform and allows parallelization of data storage and search. Scalable solutions, such as the one we developed here, will be critical components of the next-generation of tools for firearm forensics analysis. Solutions like ours will become increasingly important given the size of 3D surface topography files, the growing number of entries in scan databases, and the increasing sophistication of comparison algorithms. We are continuing development of the Nexus platform introduced here. Finally we continued to expand and support the only free X3P viewer for firearm forensics (Aim 3). This work included implementing an encrypted X3P format for the secure transmission of topography data. Through the year we continued collaboration with academic, industry, and government colleagues. We gave seven presentations at academic conferences, workshops, and crime-labs and had a full-length research paper accepted for publication in the Journal of Forensic Sciences.

Appendix

Implications for Criminal Justice Policy and Practice

Our primary impact has been the continuing development of a novel 3D imaging and analysis system with reduced cost and improved accuracy compared to existing solutions. Our work directly addresses several aims of the NIJ's Applied Research and Development in Forensic Science for Criminal Justice Purposes program. Through direct collaboration, networking, talks, seminars, and publications we have made many forensic labs (local, state, and federal), practitioners, and policy makers within the criminal justice system aware of this work. We are developing measurement and analytic techniques, grounded in mathematical science that are able to provide accurate quantitative sample comparison and database search. The ability to identify (recall) candidate minimally-marked cartridge cases can provide critical investigative leads to forensic practitioners. Our work designing a scalable computing platform for firearm forensics will ensure that database searches can be completed quickly even in the presence of growing database sizes and increasingly sophisticated comparison algorithms. Developing a distributed computing version of our algorithm allows individual labs to dramatically increase their database search speed, to share data with other labs, and to conduct large database searches. Additionally, laboratories or government agencies will not need to purchase and maintain large, in-house servers to achieve fast database search results. Finally, the availability of free X3P viewing software allows more practitioners to participate in 3D technology and the development of an encrypted form of X3P allows secure data transfer.

This work benefits the criminal justice system and their ability to present firearm identification and toolmark evidence in the courtroom. Additional impact will be made as more crime labs become aware of the work and as we continue to disseminate results. At least nine crime laboratories have had access to our technology. This would not have been possible prior to receiving recent NIJ awards. For labs that currently have 2D imaging systems, our 3D system provides a significant improvement in imaging and match accuracy. For labs that currently have competing 3D imaging systems, we feel our system offers more flexibility and transparency with respect to how the scanner works, increased resolution, improved visualization, and interpretable match score.

Figures and Tables

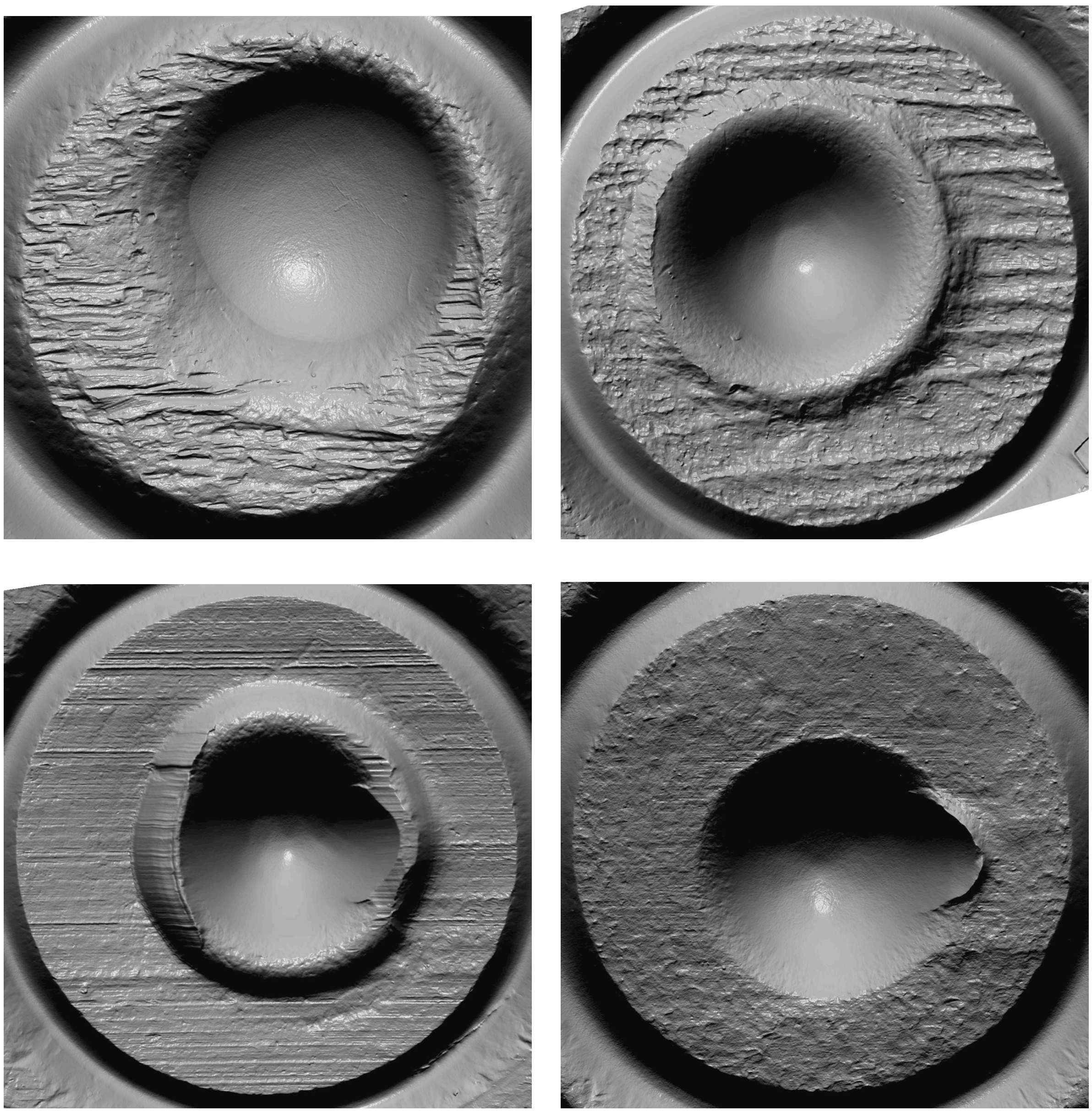

Figure 1: **Sample Sufficiently Marked Cartridge Case Scans.** (Top) Two very well marked cartridge cases and (Bottom) to less well marked cartridge cases. All four of these cartridge cases have match scores above our regular threshold. These four firearms are included in the 122 firearms considered sufficiently well marked and are not part of our minimally-marked set.

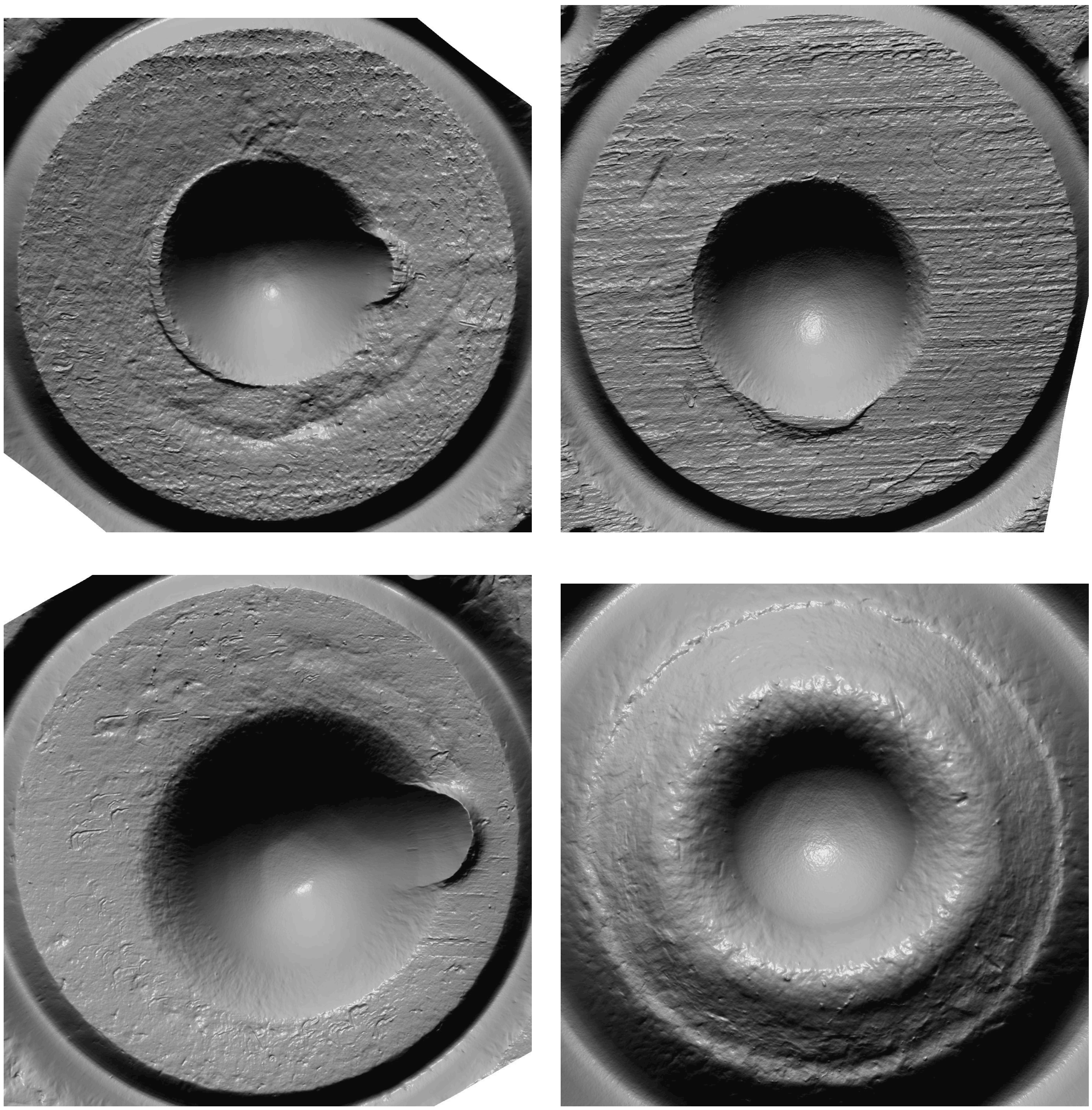

Figure 2: **Sample Minimally-Marked Cartridge Case Scans.** (Top Row) Cartridge cases from two firearms that are correctly recalled using our new matching protocol for minimally-marked cartridge cases. (Top-Left) Walther 9mm, (Top-Right) H&K 9mm. (Bottom Row) Cartridge cases from two firearms that are not recalled despite our new matching protocol. The bottom-left firearm is a 0.40 S&W Sig Sauer. The bottom-right firearm is a 9mm Beretta with the typical countersink around the firing pin aperture. The amount of surface area for the breech-face impression is essentially zero.

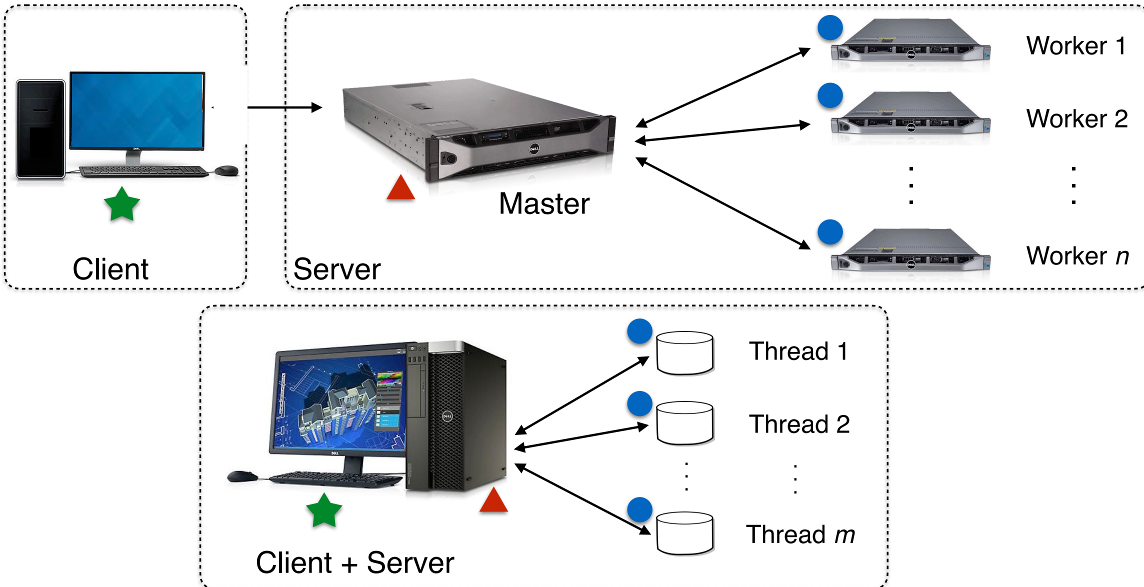

Figure 3: **Client Server Architecture.** In a distributed computing framework using a compute cluster (Top), the user runs the client software (green star) on their machine and communicates with a server consisting of a master node and several worker nodes. The master runs software (red triangle) that doles out individual jobs (blue circles) to each worker. When the entire search is complete, the server sends the results back to the client. In an alternate arrangement, all three softwares can run on the same machine (Bottom). In this setup, threads perform the work of the worker nodes. The number of threads supported by a typical desktop computer (m) is about 4-8, the number of worker nodes in a compute cluster (n) can be hundreds to thousands. The compute cluster approach (Top) can be hundreds of times faster than the stand-alone approach (Bottom).

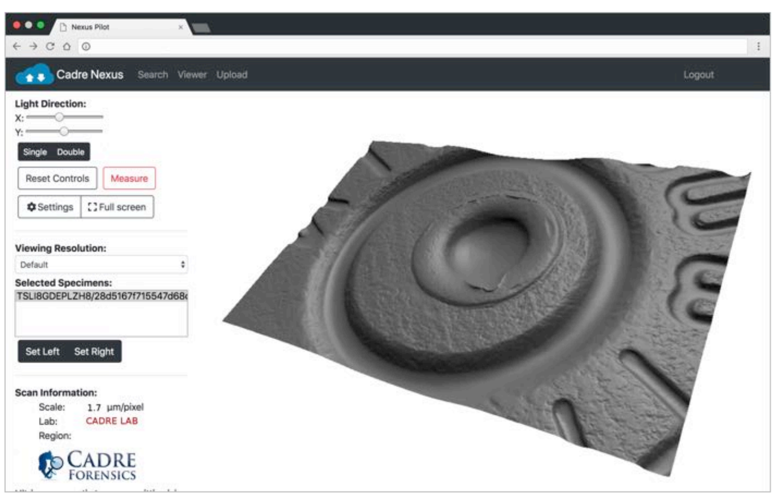

Figure 4: **Nexus Web Interface.** A user can access and view 3D surface topographies stored in X3P file format on the Nexus. Shown in the 3D web viewer which allows the user to rotate, translate, and zoom the topographic surface. The user can also adjust the position of a virtual light. Although the web-viewer is not as full-featured as our current desktop software it demonstrates proof-of-concept.

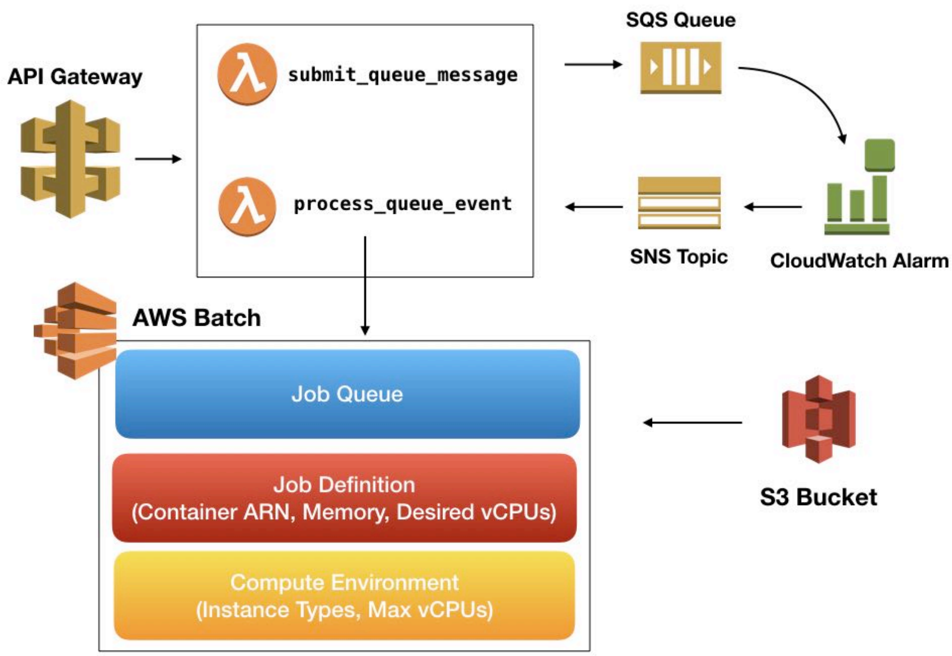

Figure 5: **Submitting a Batch Job to AWS.** A user submits a search request to the Nexus via the API Gateway. Software responding to the gateway request adds each requested comparison to a compute queue (SQS Queue). The CloudWatch service notices the request and adds comparison job to the AWS Batch Job Queue. The job specifies the required compute resources and environment.

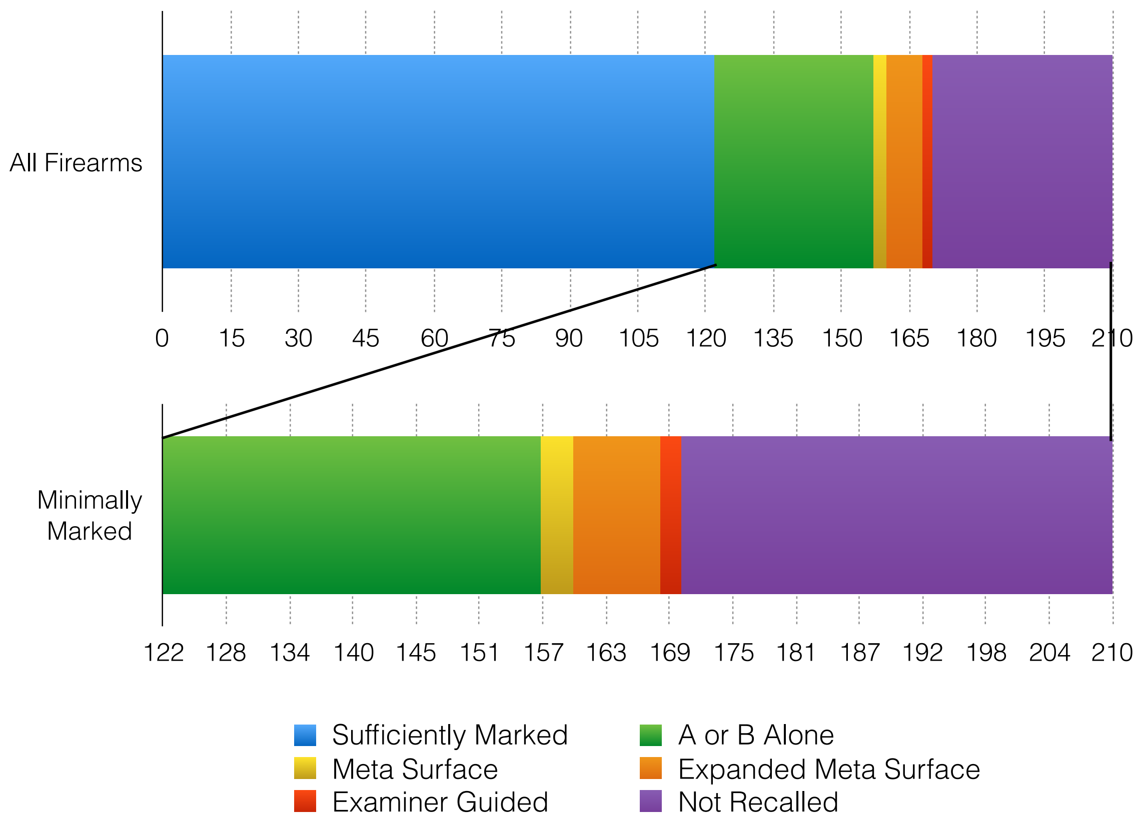

Figure 6: **Recall Results.** Recall results for all 210 firearms (top) and for just those minimally-marked firearms (bottom). (blue) 88 sufficiently marked cartridge cases with match score larger than our cutoff that are not part of our minimally marked firearm set, (green) 35 firearms with match score below cutoff but for which the top-ranked result was correct (recalled), (yellow) 3 firearms recalled using the meta-surface, (orange) 8 additional firearms recalled using the expanded feature meta-surface, (red) 2 additional firearms recalled using examiner guided masking, and (purple) the remaining 40 firearms not recalled by the new recall algorithms.

Nodes	Total Time (hrs)	Comparison Time (s)	Speed-Up
1	80.0*	64	1.0x
5	19.7	15.9	4.1x
10	10.9	8.8	7.3x
20	5.6	4.5	14.3x
40	2.9	2.4	26.7x
80	1.5	1.2	53.3x

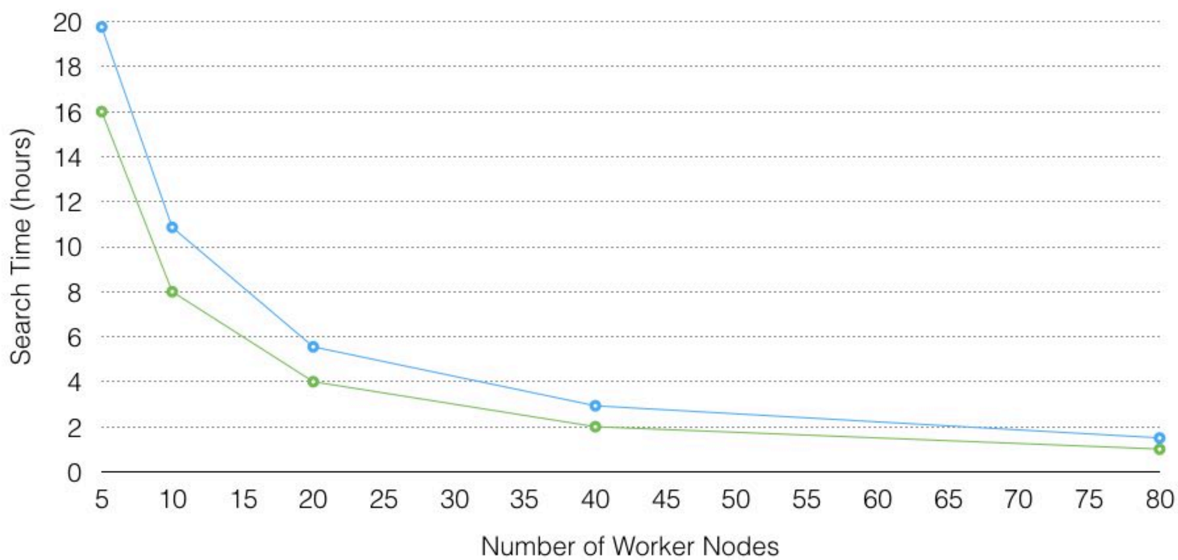

Figure 7: **Total Runtime for 4465 Comparisons.** Results are shown for the large 4465 comparison experiment. (Top) Numeric results table for 1-80 nodes. (Bottom) Graph representation for 5 - 80 nodes. Note that the runtime for one node (approximately 80 hours) is not shown. The blue line shows the search time of our implementation. The green line shows the optimal search time defined as a k-fold speedup when k worker nodes are utilized. *Note that the search time for one node is estimated based on the 200 comparison experiment.

References

[1] D. Cork, J. Rolph, E. Meleran, and C. Petrie. *Ballistic Imaging*. Committee to Assess the Feasibility, Accuracy and Technical Capability of a National Ballistics Database, National Research Council. National Academies Press, 2008.

[2] National Research Council, editor. *Strengthening Forensic Science in the United States: A Path Forward*. National Academies Press, 2009.

[3] J. DeCeuster, R. Hermsen, M. Mastaglio, and R. Nennstiel. A discussion on the usefulness of a shared European ballistic image database. *Science and Justice*, 52:237–42, 2012.

[4] P. Duez, T. Weller, M. Brubaker, R. Hockensmith, P. Duez, and R. Lilien. Development and validation of a virtual examination tool for firearm forensics. *J. Forensic Sciences*, 63(4):1069–84, 2018.

[5] J. Song. Proposed "NIST ballistics identification system (NBIS)" based on 3d topography measurements on correlation cells. *AFTE Journal*, 45:184–94, 2013.

[6] J. Song. Proposed "congruent matching cells (CMC)" method for ballistic identification and error rate estimation. *AFTE Journal*, 47:177–85, 2015.

[7] J. Song, T. Vorburger, W. Chu, J. Yen, J. Soons, D. Ott, and N. Zhang. Estimating error rates for firearm evidence identifications in forensic science. *Forensic Science International*, 284:15–32, 2018.

[8] T. Weller, M. Brubaker, P. Duez, and R. Lilien. Introduction and initial evaluation of a novel three-dimensional imaging and analysis system for firearm forensics. *AFTE Journal*, 47:198–208, 2015.

[9] T. Weller, A. Zheng, R. Thompson, and F. Tulleners. Confocal microscopy analysis of breech face marks on fired cartridge cases from 10 consecutively manufactured pistol slides. *J. Forensic Science*, 57:912–7, 2012.