GANNON TECHNOLOGIES GROUP

Using GPU Technology to Solve the Latent Fingerprint Matching Problem

July 11, 2012

GTC Express webinar

NVIDIA

Presenters

Mark A. Walch
President
The Gannon Technologies Group
Alexandria, Virginia
mwalch@gannontech.com

Y Srinivas Reddy
Founding Partner
SRIS
gsworkmail@gmail.com

The GTG/SRIS Partnership

GTG is a leading provider of content capture and biometric solutions for the Intelligence Community (IC) and law enforcement. GTG has pioneered numerous biometric technologies designed to establish individual identity from sparse or minimal sensor information.

SRIS was established through years of direct operational support to the IC and the mission need requiring exploitation of vast amounts of data. Expediting the discovery of actionable intelligence in vast computational environments is the strict visionary goal. SRIS has addressed this demand with MonsterWave and is now transitioning the platform to support multiple domains. Processing that has taken weeks in the recent past is now performed in hours. SRIS's customized framework and software application is ready to provide organizations with the solutions they demand in their mission space.

SRIS and GTG relationship:

SRIS and GTG began their relationship six years ago in support of the intelligence communities research and development. Through years of growing together and understanding customer problem sets, SRIS and GTG have now partnered with the goal to fulfill an operational community need that has remained unsolvable.

Overview of Today's Presentation

Part 1: **Discussion of a Novel Technology for Matching Latent Fingerprint**

Mark A. Walch

Part 2: **Application of GPU Technology to the Latent Fingerprint Solution**

Sri Reddy

Part 1: Discussion of Latent Fingerprints

By Mark A. Walch

The Problem

Latent fingerprints represent a significant challenge to automated fingerprint matching and may latent fingerprint are not matched because they lack sufficient information for conventional matching methods.

Significance

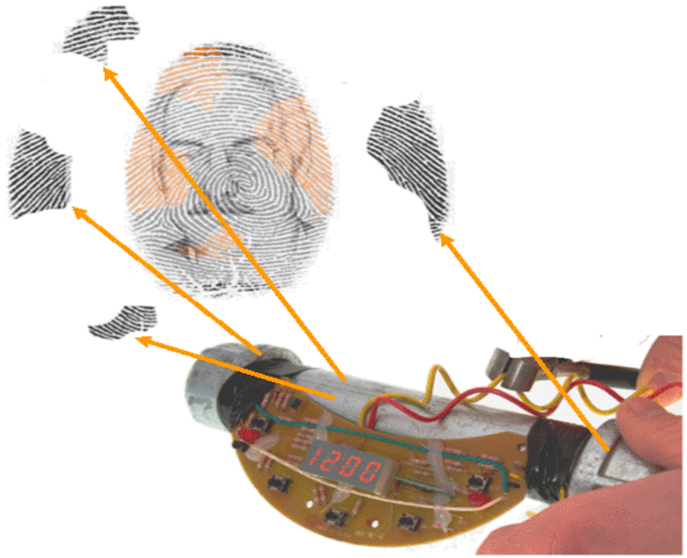
Latent fingerprints are important for two reasons:

1. Millions of latent prints have been collected or are available for collection but they cannot be automatically matched by conventional methods.

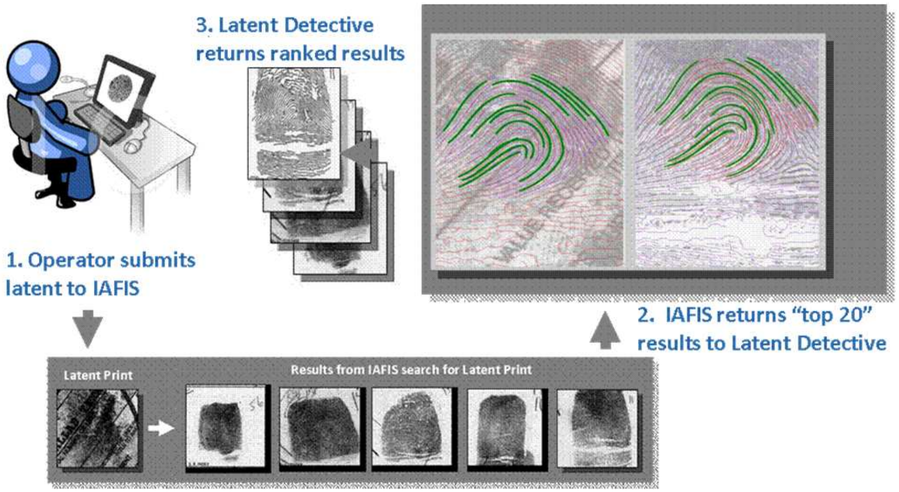
2. Large databases such as the FBI's IAFIS system will only return 20 matches to a latent print that is submitted as a search query. The number 20 is what can be reasonably expected for a fingerprint examiner to manually process. As the database grows larger, this limitation has statistical consequences.

Objective for the Current Research

1. A "lights out" latent print matcher that can make the millions of available prints searchable without human intervention.



2. The "IAFIS Afterburner" that will permit users of the IAFIS system to handle more than 20 returned cases.



A Solution

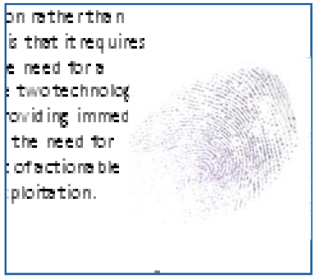
Gannon Technologies ("Gannon") has developed a method for meeting the challenge posed by latent prints using ridge flow as an alternative to conventional minutiae matching. A matching approach focusing on ridges benefits from eliminating the need to detect minutiae which can be difficult even for human experts.

Furthermore, a ridge based approach can find useful identity information in sparse prints with insufficient minutiae for conventional matching.

A Fingerprint Primer

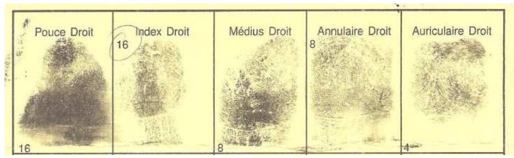
The “fingerprint world” is divided into two types of prints: “Latents” which are prints left by an individual and “Tenprints” which are captured directly from the finger.

“Latents”



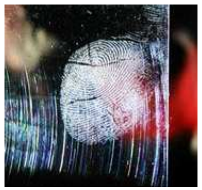
Multiple finger fragments

Paper and absorbent media



Poor quality tenprints

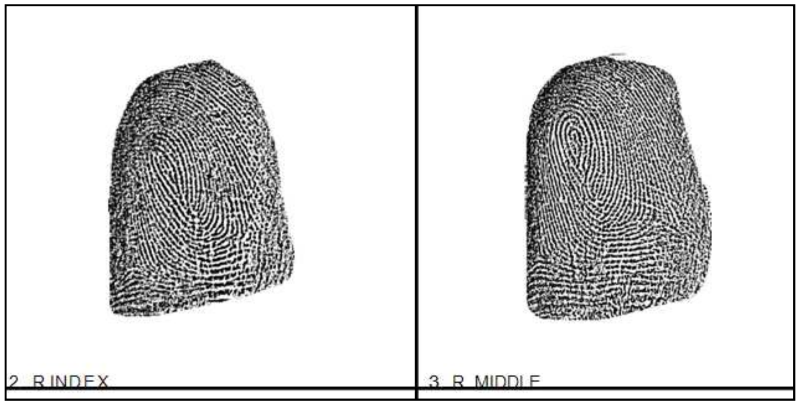
Shell Casings/IEDs



“Tenprints”

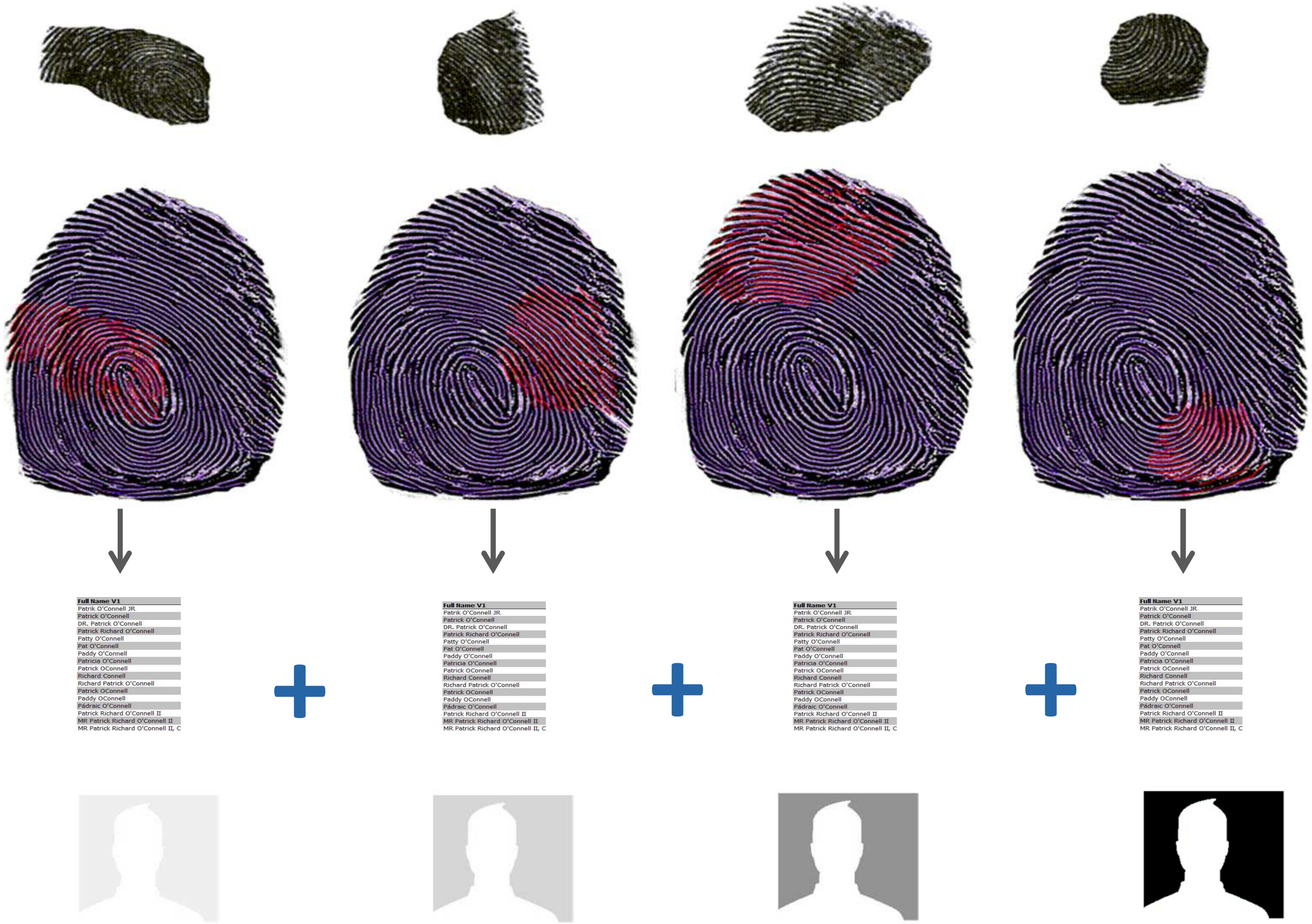
Livescan

Fingerprint Card



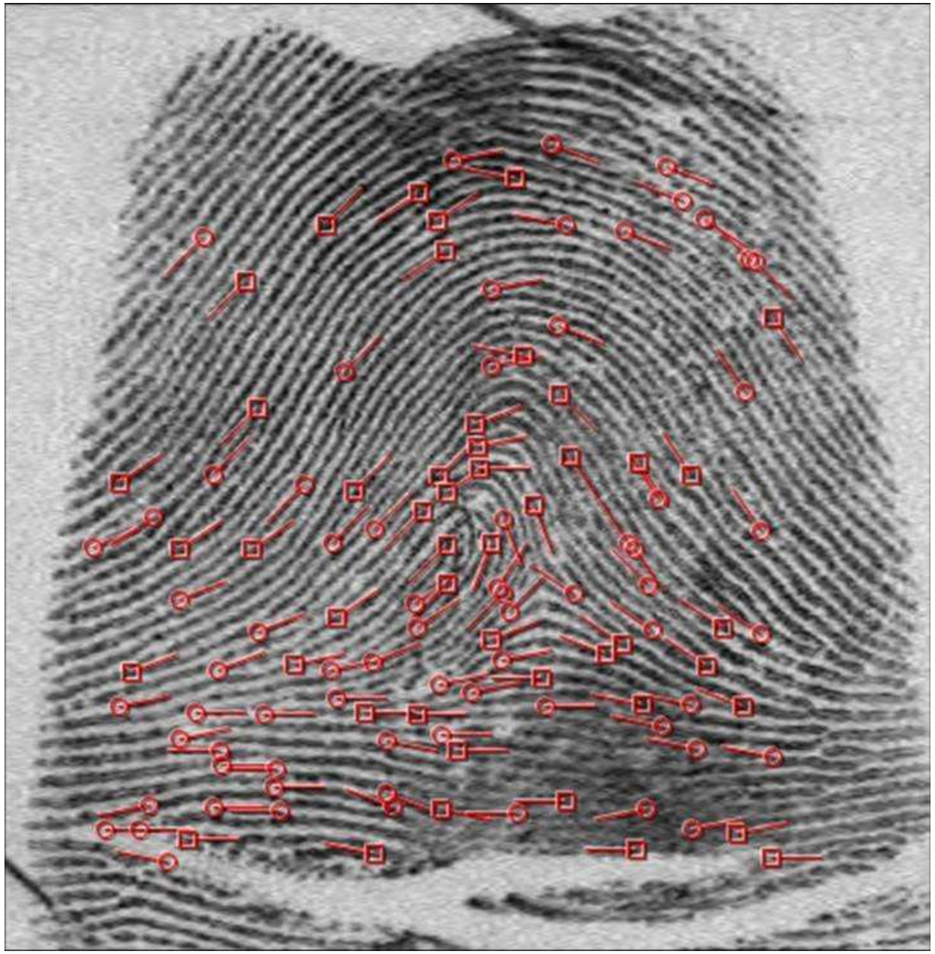
Identity in Fingerprint Fragments

The problem is further compounded that many latent prints lack sufficient information individually to support identity. However, latent fingerprints can also be “fused” to create a full picture of identity from multiple individual “parts”.



The Issues with Latent Fingerprints

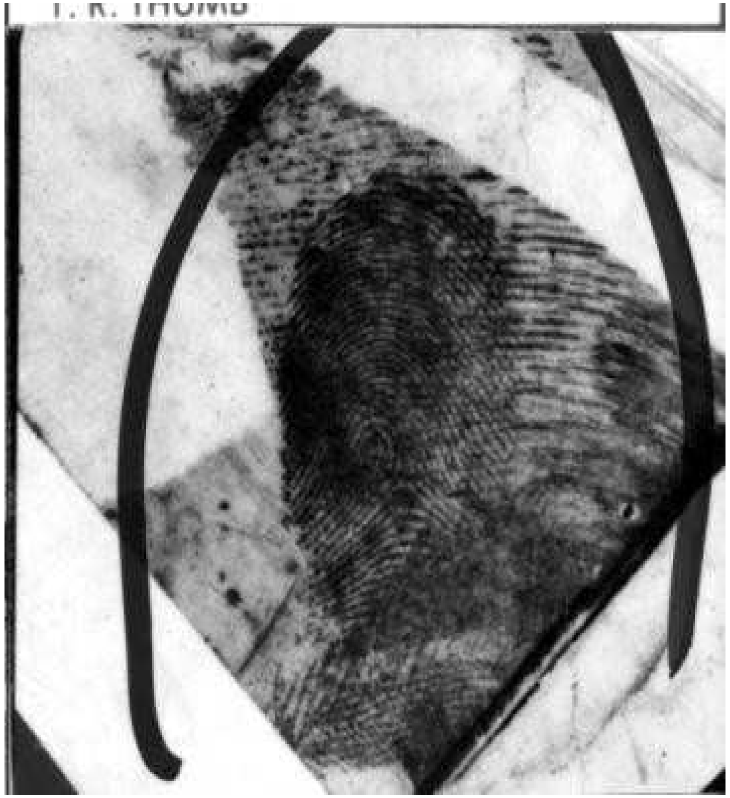
Traditionally, minutiae (bifurcations and terminations of friction ridges) are the mainstay of fingerprint-based identification.



However, latent prints often lack sufficient minutiae to permit minutiae-based identification—particularly for prints captured from paper and other absorbent surfaces.

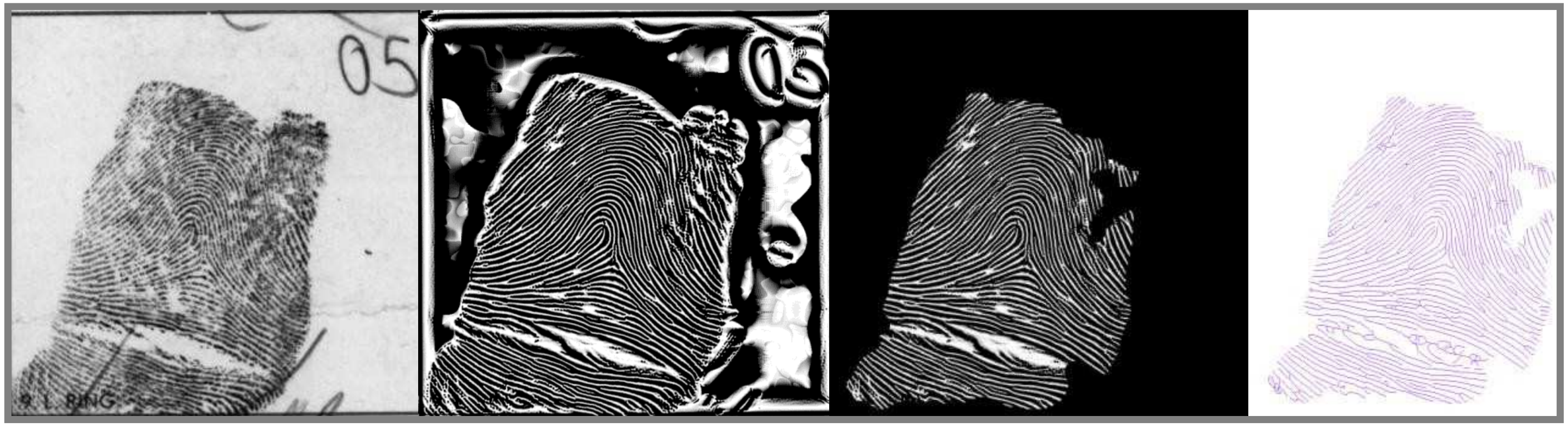
Identifying Latent Fingerprints

The image on the left shows a latent fingerprint as it is captured at a crime scene or sensitive site. The image on the right shows a corresponding reference print from the same finger. Following is a step-by-step description of how the latent and reference can be matched.



The Reference Print Database

Ridge-based content can be captured from fingerprints by transforming the original print into a “high contrast” image which is then masked and “thinned” into a “skeleton” image delineating ridge flow.



Original reference fingerprint

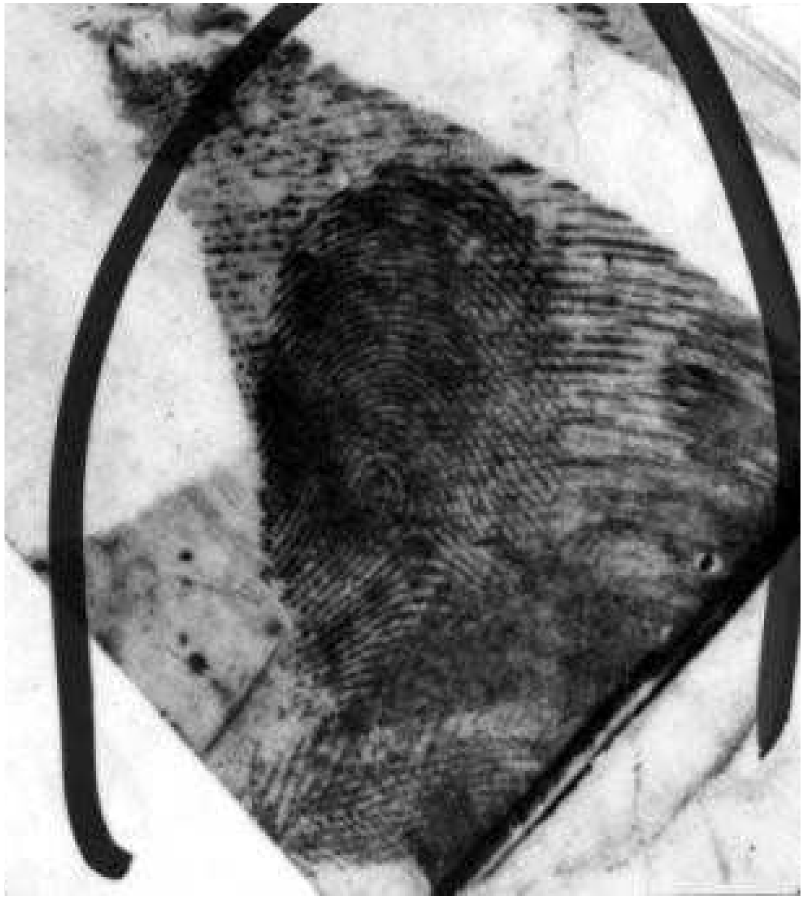
High contrast representation created by detecting periodic pixel patterns

High contrast image masked to remove all areas not showing “regular” patterns

Skeletal image produced by thinning all ridge lines to single pixel width

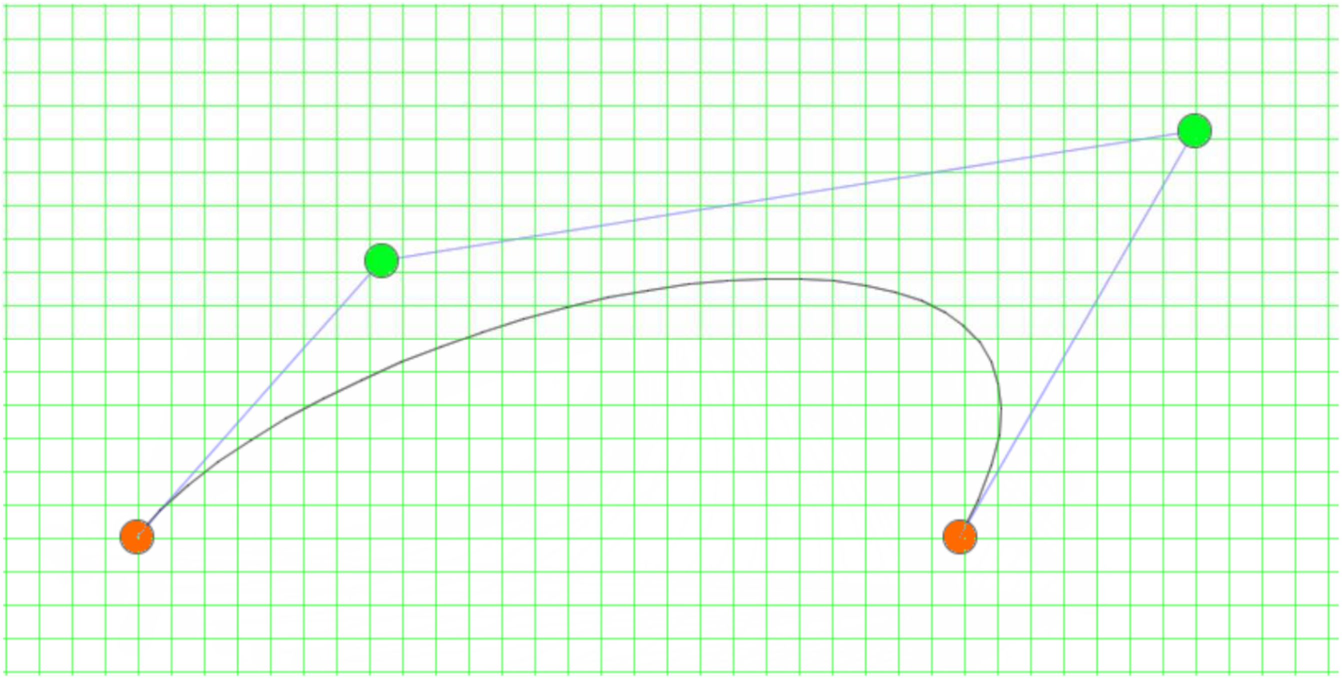
The Latent Print

The latent print is separated from its background and converted into a “trace” image either by image processing or tracing depending on the amount of noise in the image.



Ridge-Specific Markers: Capturing Curve Shape

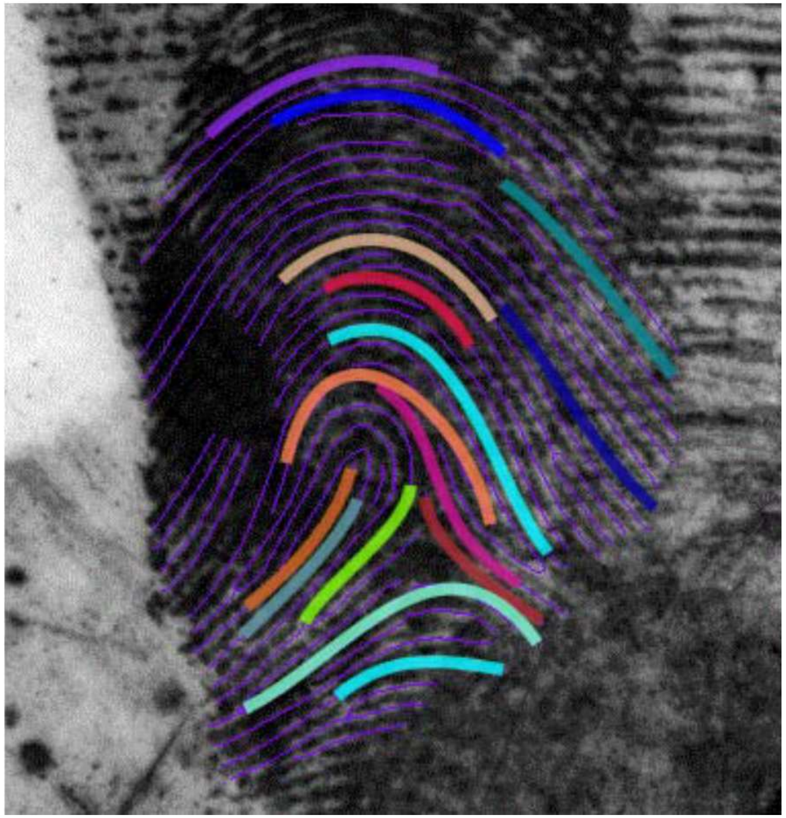
A tool can be appropriated from the world of handwriting recognition that offers a means of generating Ridge Specific Markers. This tool takes the form of Bezier-based curve descriptors. Using Bezier descriptors, any simple curve can be described through 4 points: 2 endpoints and 2 control points.

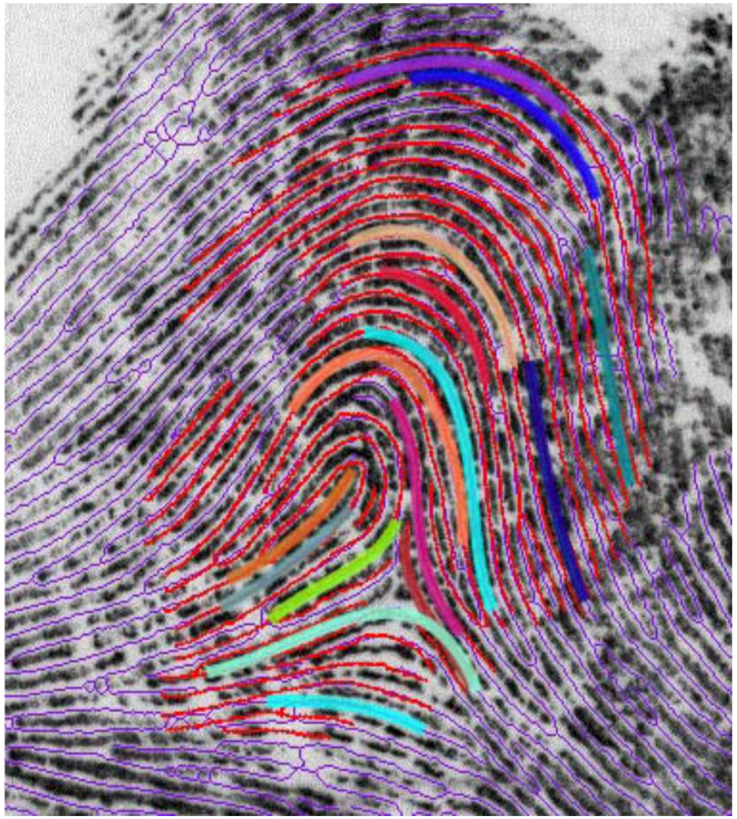


Curve and Bezier descriptors

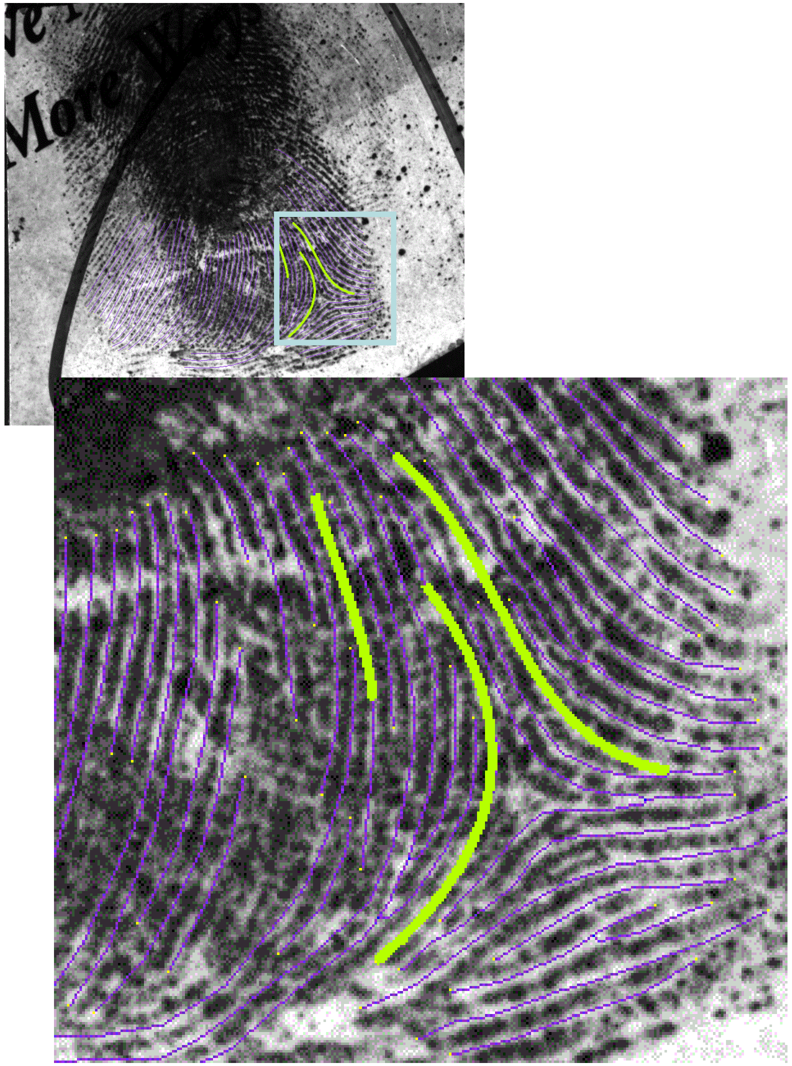
Latent-to-Reference Matching

Ridge-based matching involves finding a subset of corresponding ridge sections common both the latent and reference prints

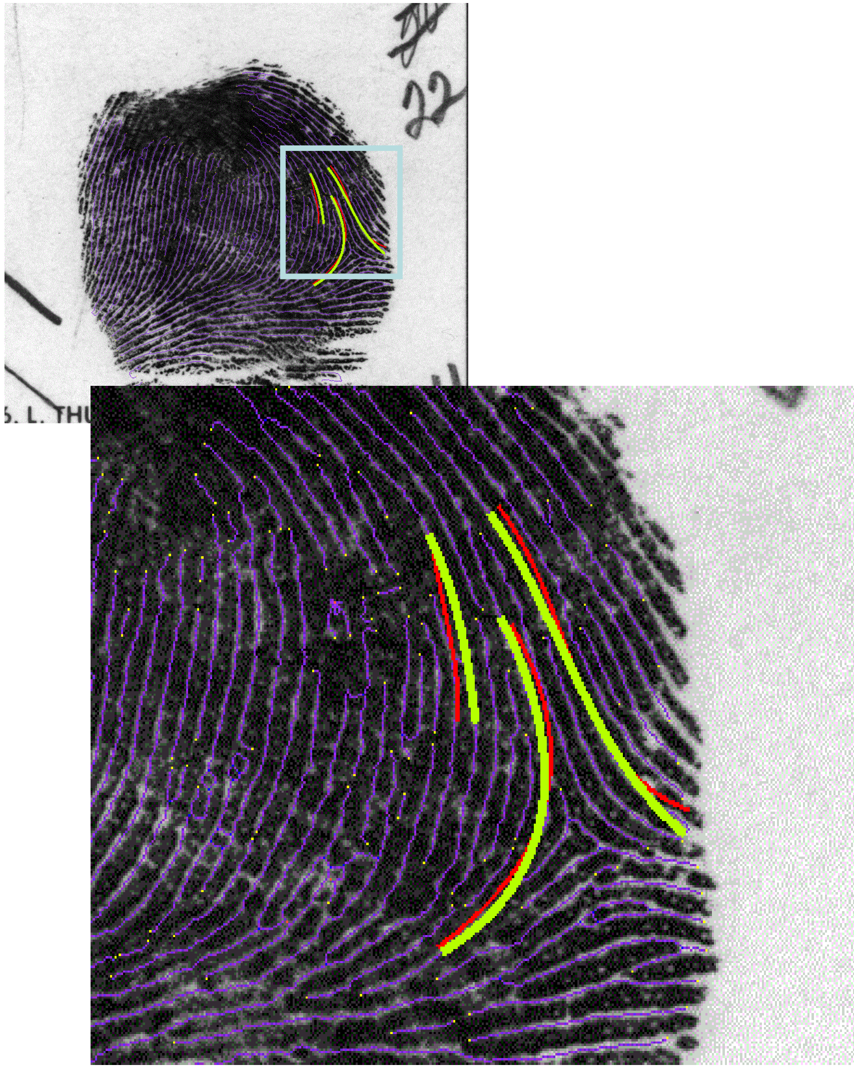




Step-by-step Curve Matching

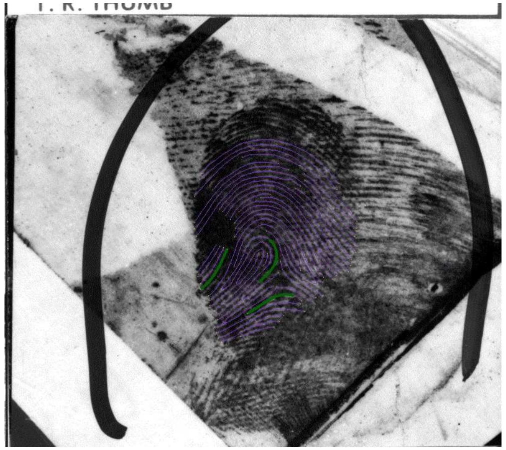


Latent Fingerprint

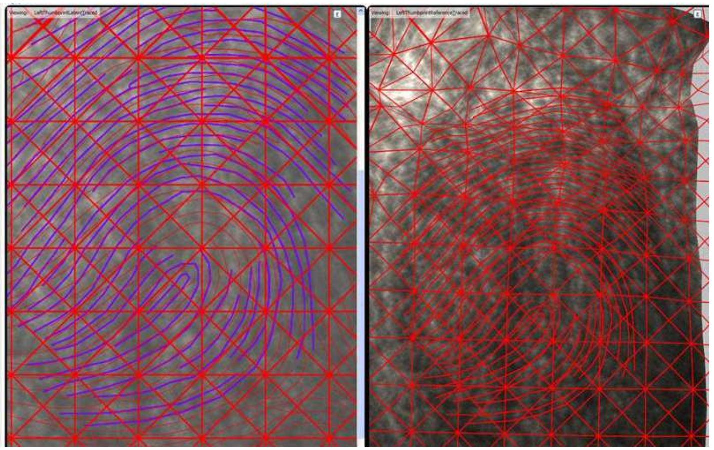


Reference Fingerprint

Constructing the Transform



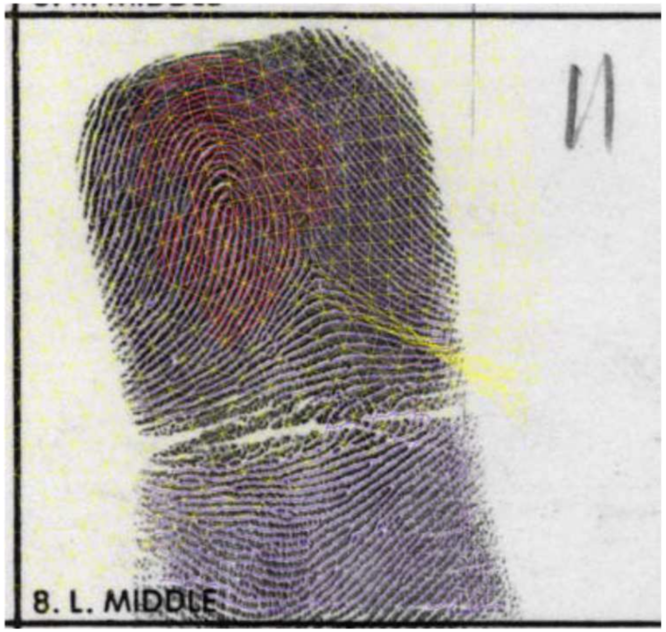
Latent Fingerprint



Latent/Reference Transform

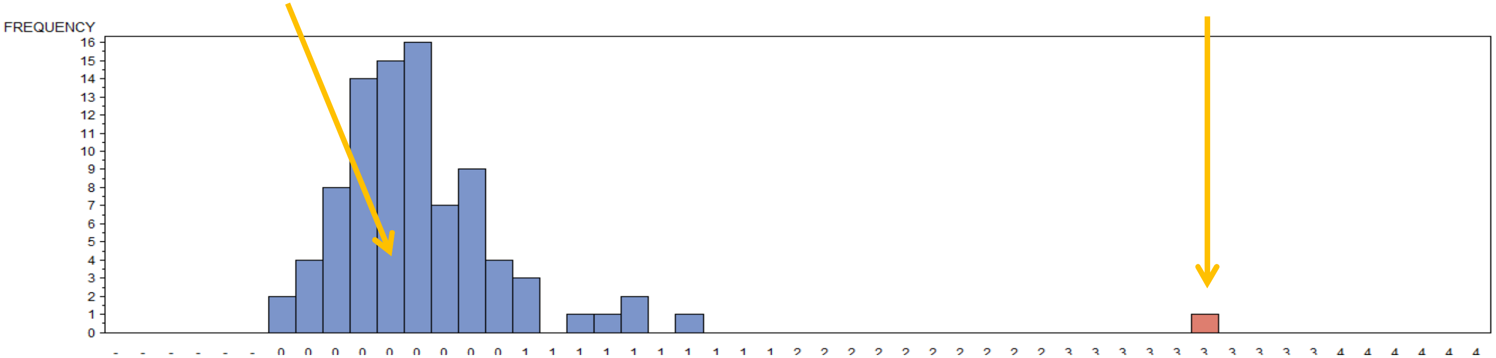
Latent Fingerprint Mapped into Reference

Scoring the Result



Incorrect Match

Correct Match



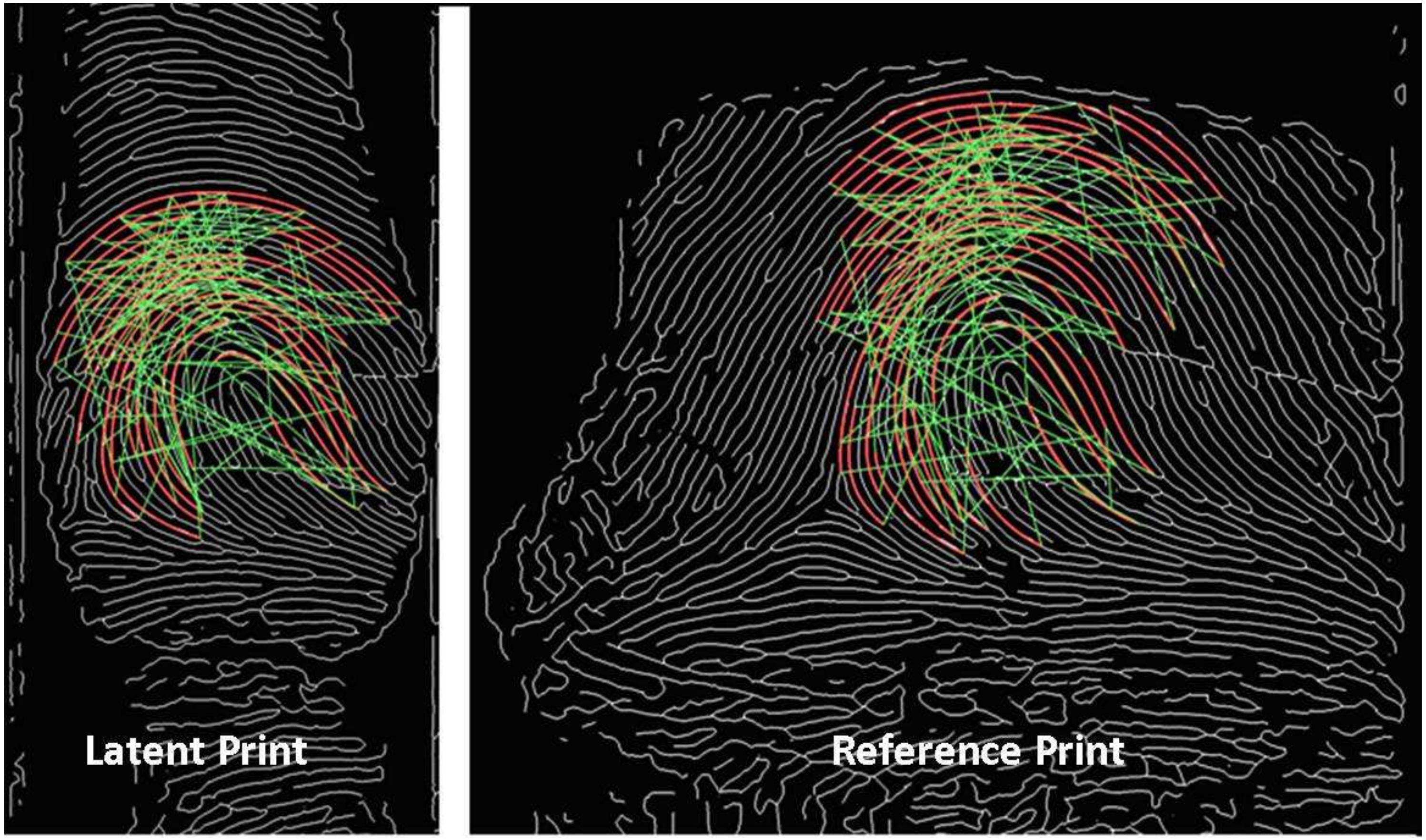
Distribution of Scores

The Challenge

Gannon's ridge flow approach to examination of latent fingerprints creates a monumental challenge of having to perform possibly trillions of calculations for indexing and a desired return when matching latents to a large reference set. To accomplish this on CPUs has become time and cost prohibitive.

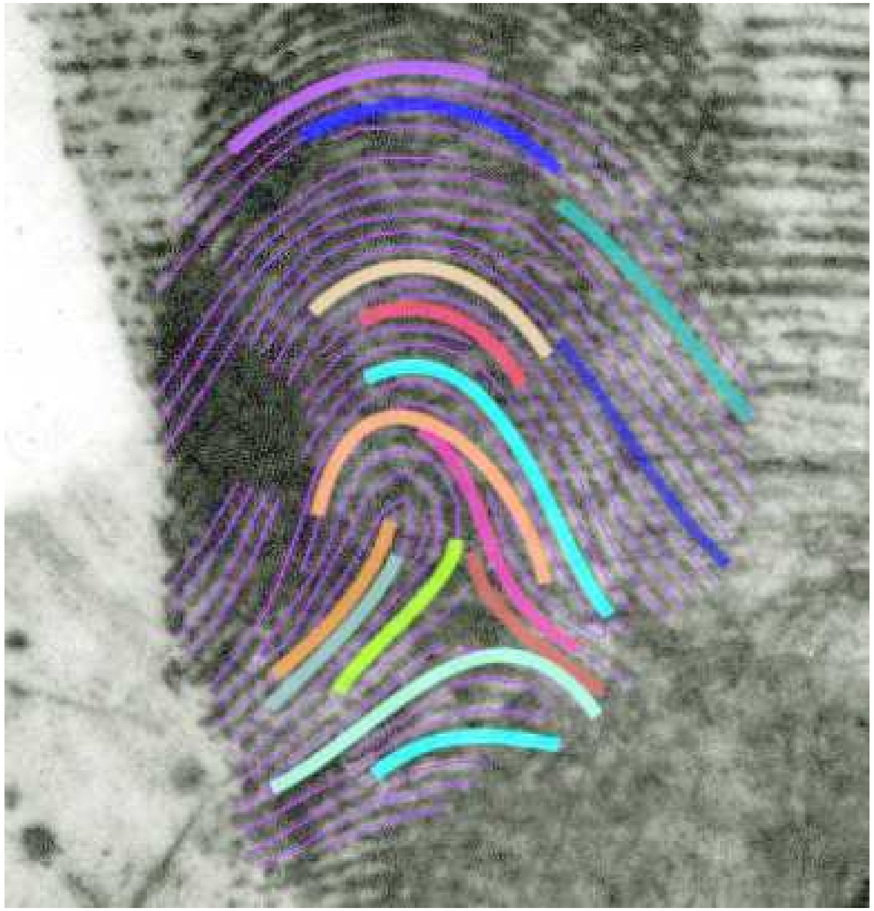
Computational Complexity

Using multiple curve-based markers, several reliable points can be matched between prints ultimately leading to the correct matching of the prints. These points collectively possess the “power of minutiae”.

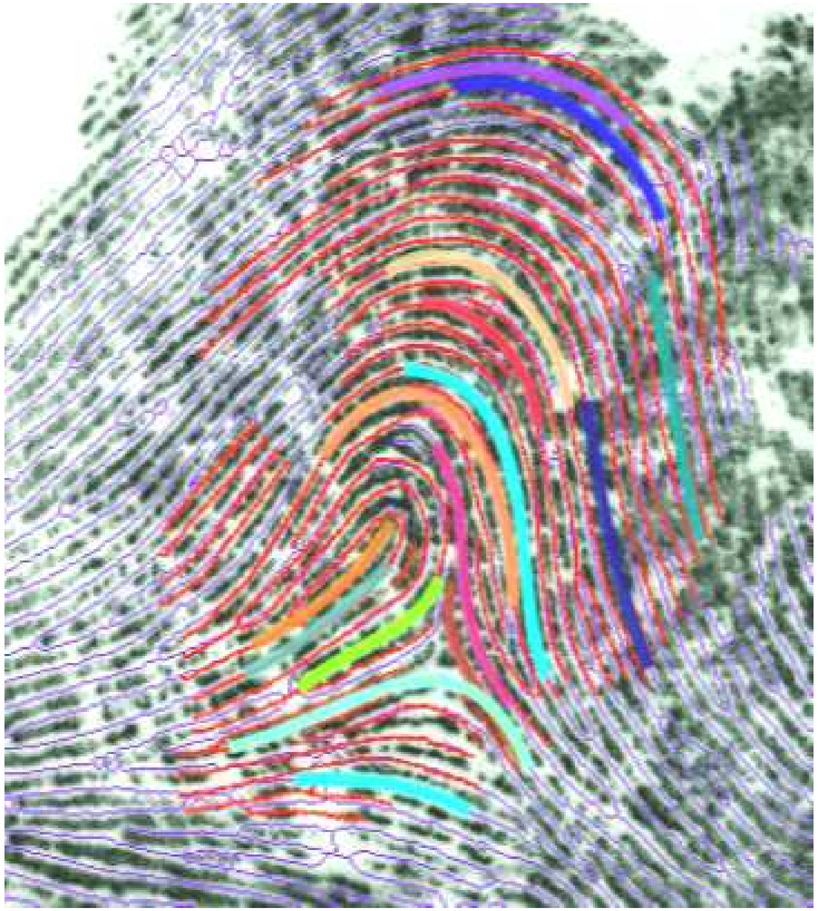


Computational Complexity: Strategy 1

The first strategy for matching is to select ridge "tuples" each containing three Bezier-based ridge segments.



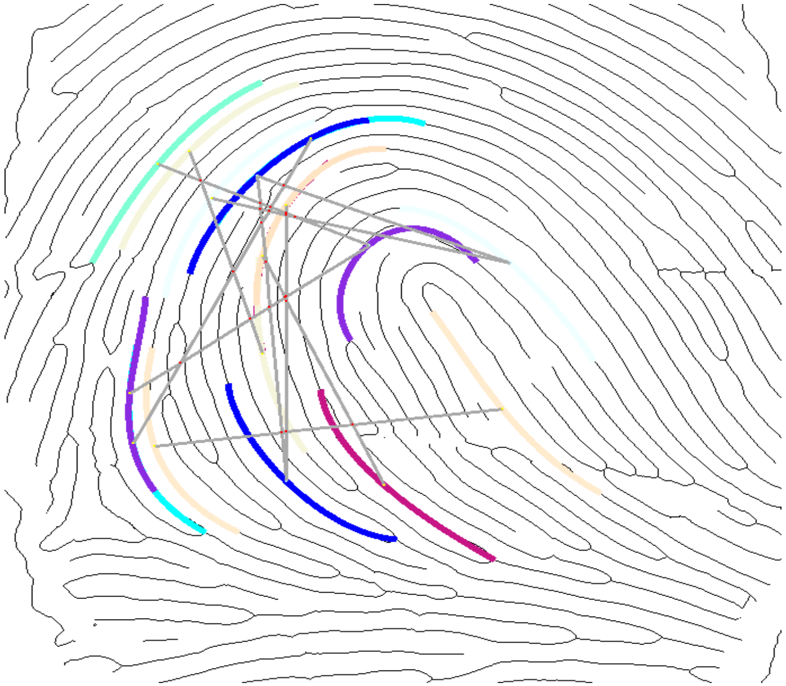
Sample set of Ridge tuples in latent print



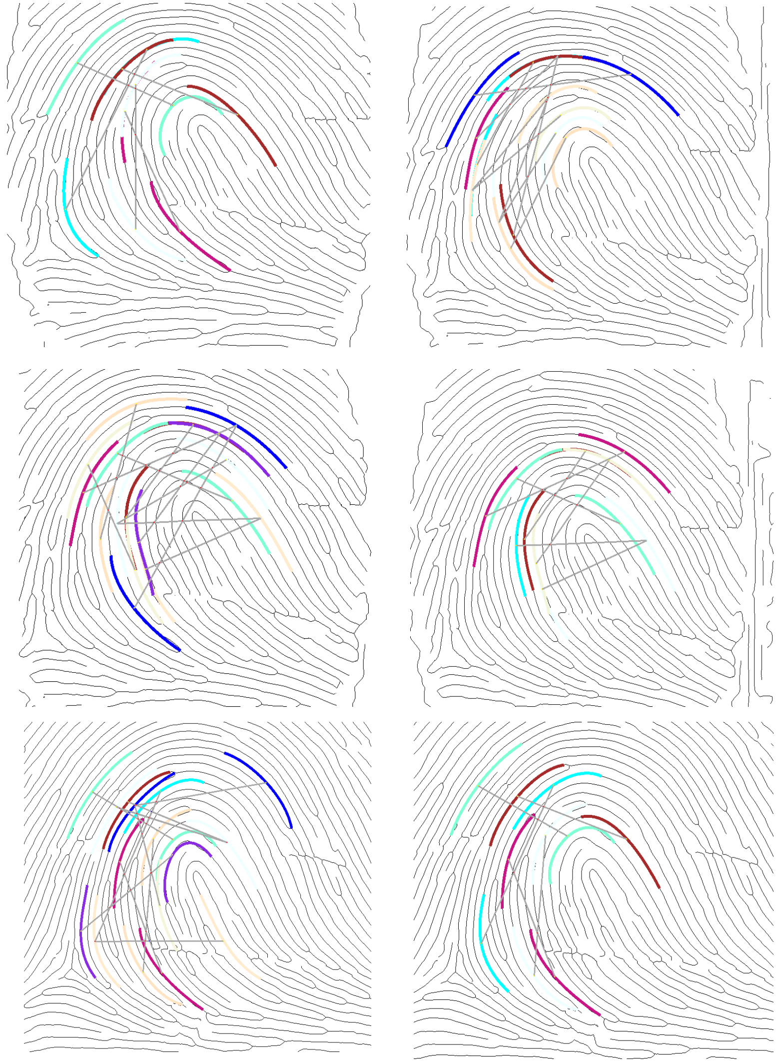
Corresponding tuples in reference print

Computational Complexity: Strategy 2

The second strategy for matching is to select larger groups of ridges within certain proximity and generate a graph with an encoding for its topology and geometry to be matched between a latent and corresponding reference prints.



KEY35;2.4.228.0.0.64.116.0.0.24.69.0.0.66.2.2.0.16.130.
2.0.192.64.32.4.8.0.0.18.64.0.4.0.0.0.0.0.0.0.32.3.



Each graph yields a unique key.

Part 2: The Application of GPU Technology

By Sri Reddy

A Novel Approach

With the introduction of GPUs, much expanded processing capacity numbers have become attainable, but as data scales in size exponentially, an equivalent problem begins to arise. A new approach to address this mission need has been created with these vast numbers in mind, and the ability to scale almost infinitely.

GP-GPU Operational Testing Findings

- A GP-GPU cluster has been used in an operational testing environment with similar computational demands.
- The system with 16 GPUS performed 160 Trillion calculations per GPU per hour (2.560 Quadrillion calculations per hour)
- Data processing previously taking **8 days** has been reduced to approximately **one hour** on the GPU cluster.

Metrics

- Latent prints are fractions of a full print, a one-to-many comparison entails breaking down both the full ('reference') print and the latent into their Bezier curves, and seeking overlapping sets.

- The factors that affect the number of curves produced include:
  - The size of the print
  - The size of the curvature segment selected
  - The number of curves sampled from ridgelines using an "overlapping stepping" method.

- A given latent fingerprint has anywhere between ~50 to ~1000 Bezier curves associated with it, and each reference print has anywhere between ~10,000 to ~20,000.

Metrics

- Current Method: To process of a given latent print (LP), a sample of 3-tuple Bezier curve "sets" from LP is created.
- GPU Method: Every combination of 3-tuple Bezier curve set from LP is created.
- Current Method: Each one of these 3-tuple sets need be compared to find a "best possible matching" 3-tuple Bezier curve set in every reference print in the corpus (greedy algorithm to find the optimal bezier curve set)
- GPU Method: Every combination of 3-tuple Bezier curve set from RP'
- For one full print to be compared against a database of 100 million reference sets (each person has 10 finger prints) the numbers are: 20k bezier X 10 prints X 100M sets= 20 trillion beziers)

Realistic Metrics utilizing GPU

- Combinations for latent print with 500 Beziers
  - $_{500}C_3$ = 500! / 3!(500-3)! = 500! / 3*2*1(497)! = 500*499*498 / 3*2*1 = 124251000/6 = **20,708,500**

- Combinations for Reference print with 18000 Beziers
  - $_{18000}C_3$ = 18000! / 3!(18000-3)! = 18000! / 3*2*1(17997)! = 18000*17999*17998 / 3*2*1 = 5831028036000/6 = **971,838,006,000**

- Compare and calculate similarity score
  - 20,708,500 latent combinations with 971,838,006,00 reference combinations

MonsterWave Specifications

- GPU/CPU Management and Monitoring Capability
  - GPU Utilization
  - Memory
  - Operating Temperatures

- Server
  - Send data to MonsterWave via socket (.NET, Java, C++) or Storm Thrift Client

- Job Scheduler
  - Schedules Jobs and queues data for GP-GPU Processing based on real-time GPU status

GPU Strategy

MonsterWave Design

Client Sends Latent Fingerprint to MonsterWave Platform

Via Socket (C++/C#/Java)

Via Thrift Client

MonsterWave (MW) Platform Server (Rocks O/S)

STORM (distributed fault-tolerant and real-time computational system to guarantee delivery of Latent data)

MW Schedules Jobs and queues data for GP-GPU Processing

GP-GPUs with FusionIO's IOMemory Device

Hyperdex (searchable distributed key-value store for Reference data)

CUDA/C++ Algorithms for comparing and scoring Latent & Reference data

Methods for Latent Fingerprint Analysis

- Pre-Process Reference data
  - Store 3-tuple combination Bezier sets in Hyperdex on FusionIO ioDrive
  - Exchange speed for disk space (go Ricky Bobby!)
- Utilize Non-Volatile Memory (prevent re-loading of Reference data) using FusionIO ioMemory
  - Each node loads specific set of reference data from FusionIO ioDrive
- MonsterWave receives latent fingerprint
  - Quickly generates combinations of 3-tuple Bezier sets

Methods for Latent Fingerprint Analysis

- Processing
  - MonsterWave sends STORM messages to various nodes in cluster for processing
  - Node receives message and passes data to processing server on the node
  - Server calculates Similarity Score

- Post-Processing
  - All nodes send results to single collection node
  - Best Latent triplet with best matching Reference triplet is derived
  - Data is sent to SQL Server using the SQL API++ (FreeTDS)

MonsterWave Hardware

- **The Dell PowerEdge R720**
  - Dell's latest 2-socket, 2U rack servers that is designed to run complex workloads using highly scalable memory, I/O capacity, and flexible network options. The R720 can readily handle very demanding workloads spanning multiple domains, such as data warehousing, e-commerce, virtual desktop infrastructure (VDI), and high performance computing (HPC) as a data node.
- **NVIDIA Gemini PCI X3.0**
  - A new graphics architecture that will replace the Fermi architecture. The new architecture is expected to have significantly increased computational efficiency and 3-4 times higher double-precision floating point performance-per-watt. (Research indicates Performance of over 6TFLOPS Single Precision per GP-GPU)
- **FusionIO ioDrive2 Duo**
  - Integrated within the server to offer advanced performance and scalability across application and databases with minimized latencies. For example, a large graph (approximately 6,871,900,000 nodes) can be processed and stored on a single system. Scalable to utilize FusionIO's direct-attached high performance ioMemory technology. The Fusion ioDrives provide low latency access to graph nodes and edges, enabling a unique alternative for data intensive computing.

MonsterWave Software

- **STORM**
  - The Storm clustered environment can handle large number of transactions in the biometric context utilizing CUDA algorithms running on clustered NVIDIA GPUs and access reference data stored on the HyperDex cluster.
- **NVIDIA CUDA**
  - CUDA language can be utilized for running algorithms (kernels) on the NVIDIA Gemini for graph processing: generating identity hash and comparing with the object database. The NVIDIA lends itself to creating and traversing identity graphs.
- **HyperDex**
  - A searchable distributed key-value store that is magnitudes faster than other systems. The decision to use a HyperDex cluster for the storage and retrieval of identified hashes was based on its expedient search, fault tolerance, consistency, and scalability. The efficient search capability is provided via a new sharding technique called hyperspace hashing that is coupled with a novel replication protocol called value-dependent chaining.

Next Steps

- **Test on NVIDIAs Petaflop Supercomputer**

Questions

- **Did someone say: Make It Go Faster Ricky Bobby!**