Liveness: A Score Mapping Methodology for Usability and Thresholding

Terry Riopka, Senior Director of Research
Jens Peter Hube, Chief Scientist

Motivation

- Most biometric algorithms return similarity or classification scores
- Decision boundary problems – require setting of operational thresholds
  - ➢ critical to determining balance between false rejections and false acceptances
- Proposal: apply concept of FMR-based score mapping (first proposed by Griffin, Hube, and Mahlmeister for use with the BioAPI standard in 2004) to liveness
  - ➢ enables setting a direct (meaningful) correspondence between thresholds and expected operational error
- Lack of theoretical justification, but empirically useful

Overview

Motivation

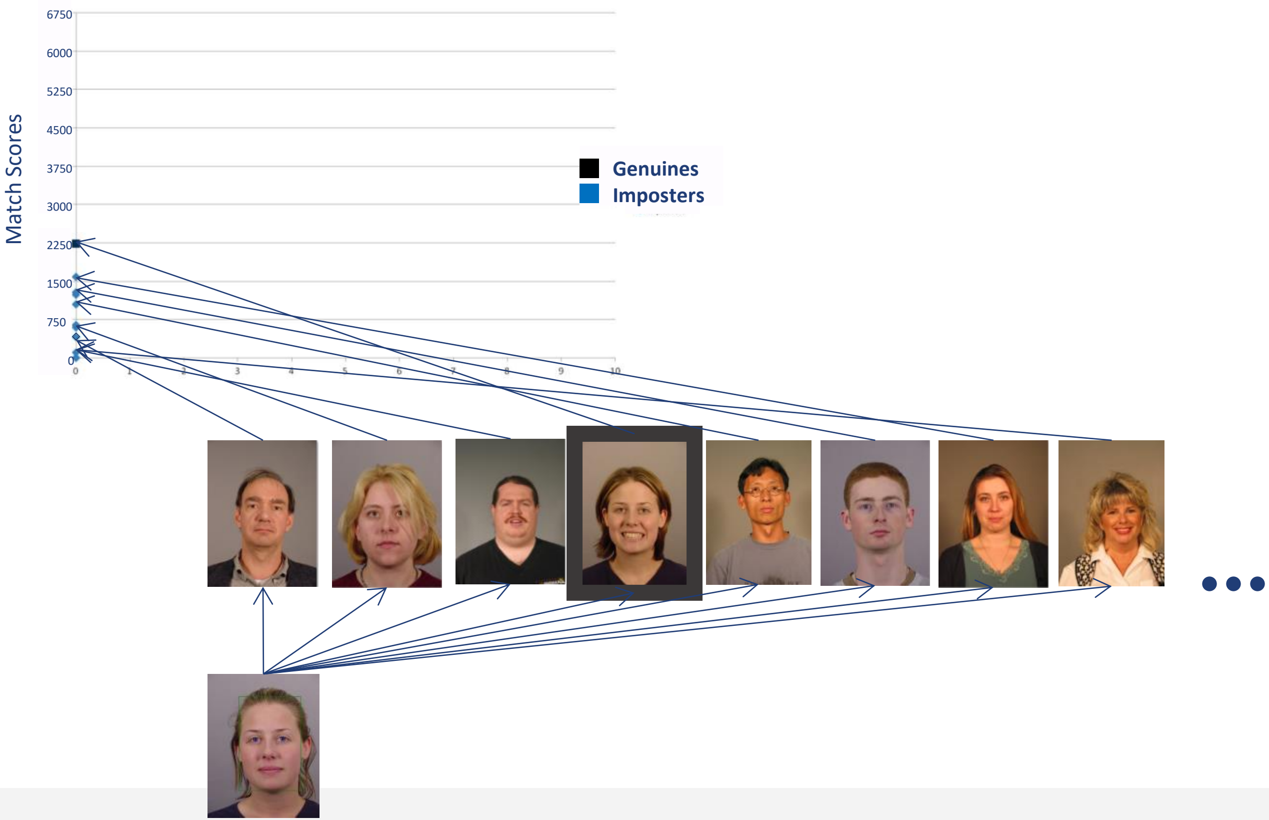
Performance Metrics in Matching – FMR

Introduction to FMR-based Score Mapping for Matching

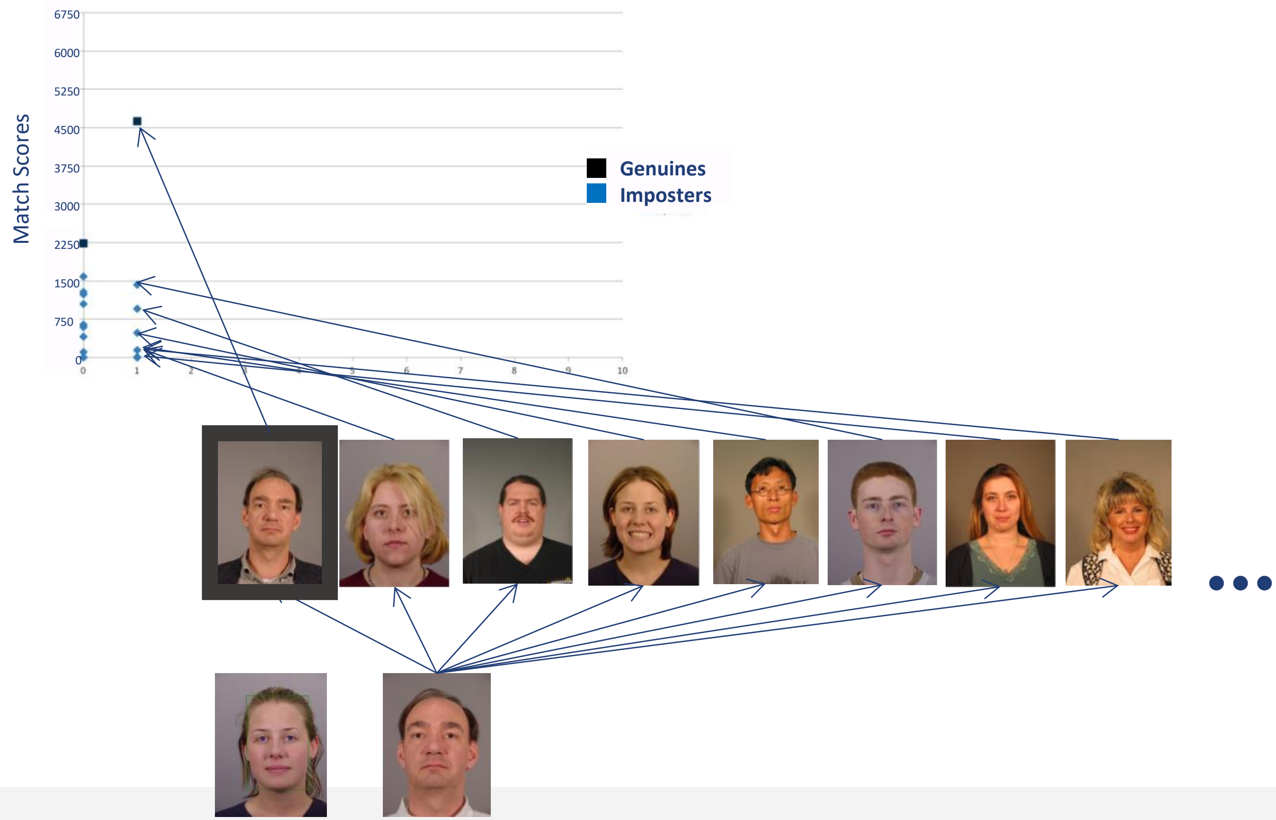
BPCER-based Score Mapping for Liveness

Science Forward: The Intuition Behind BPCER-based Score Mapping

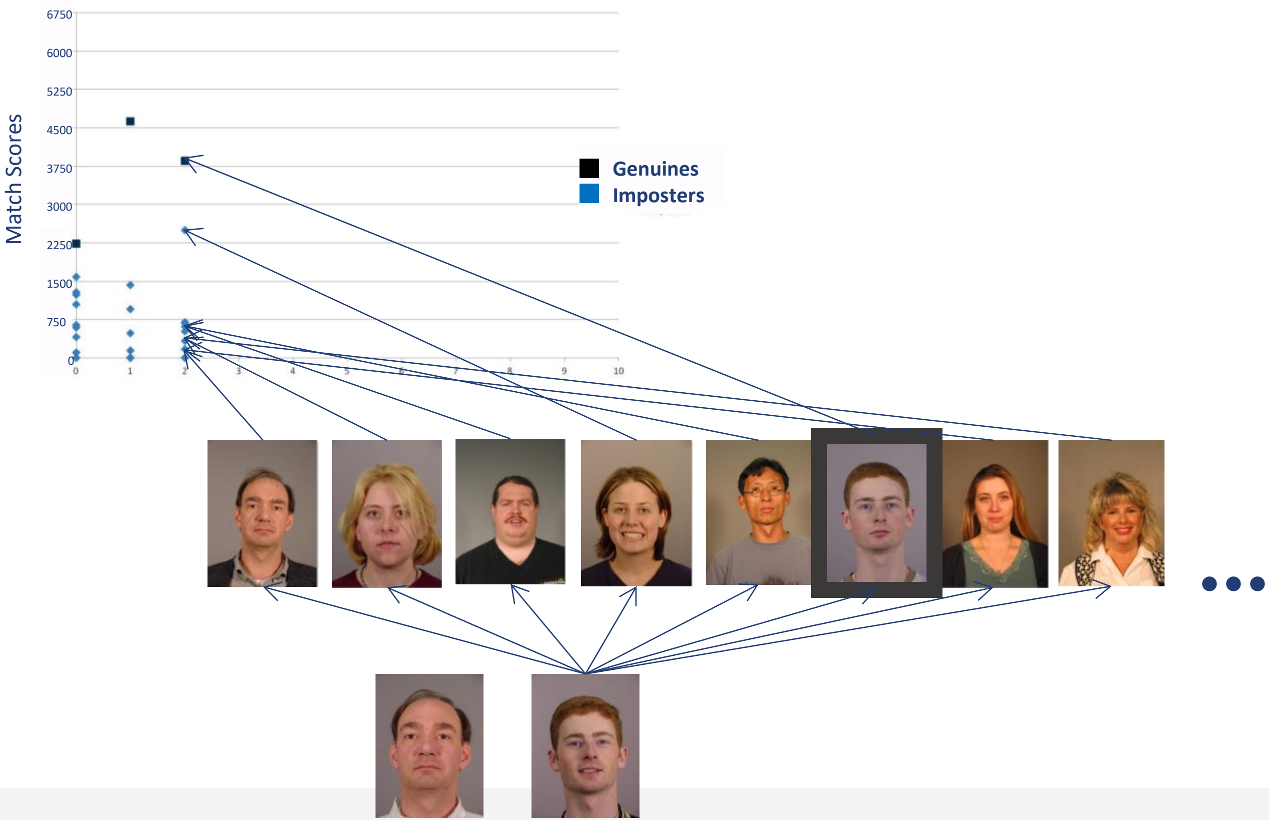
Introduction to Biometric Performance Metrics



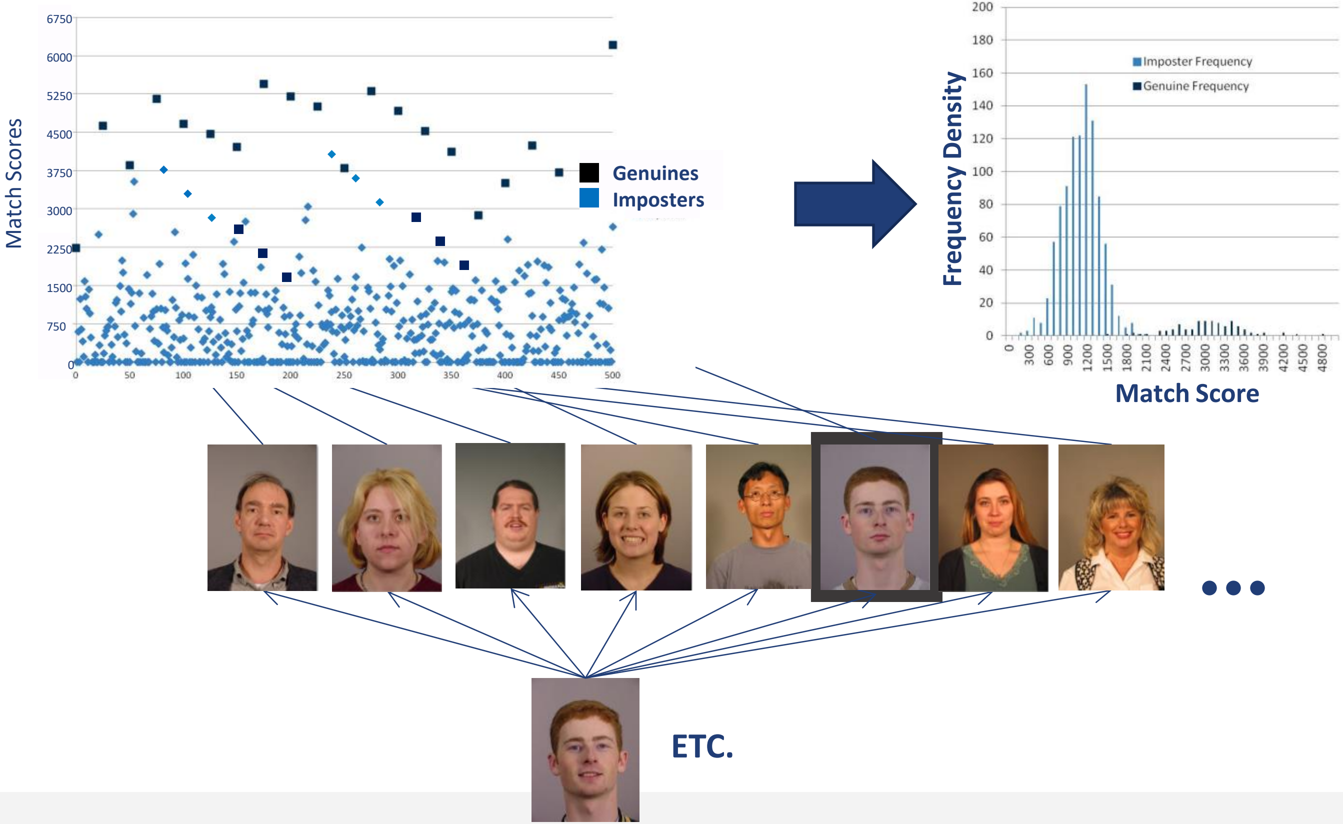
Introduction to Biometric Performance Metrics



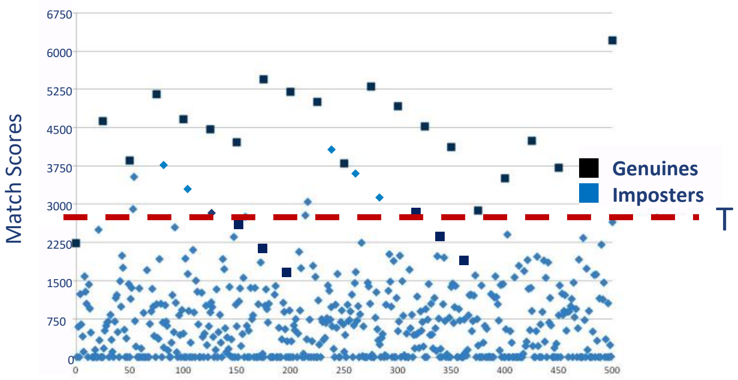
Introduction to Biometric Performance Metrics

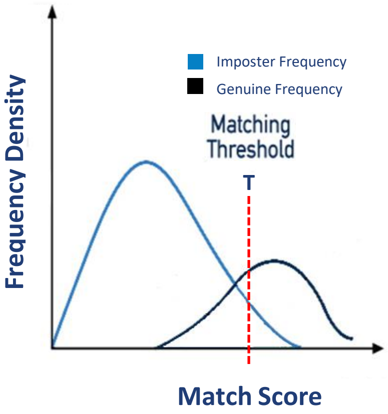


Introduction to Biometric Performance Metrics



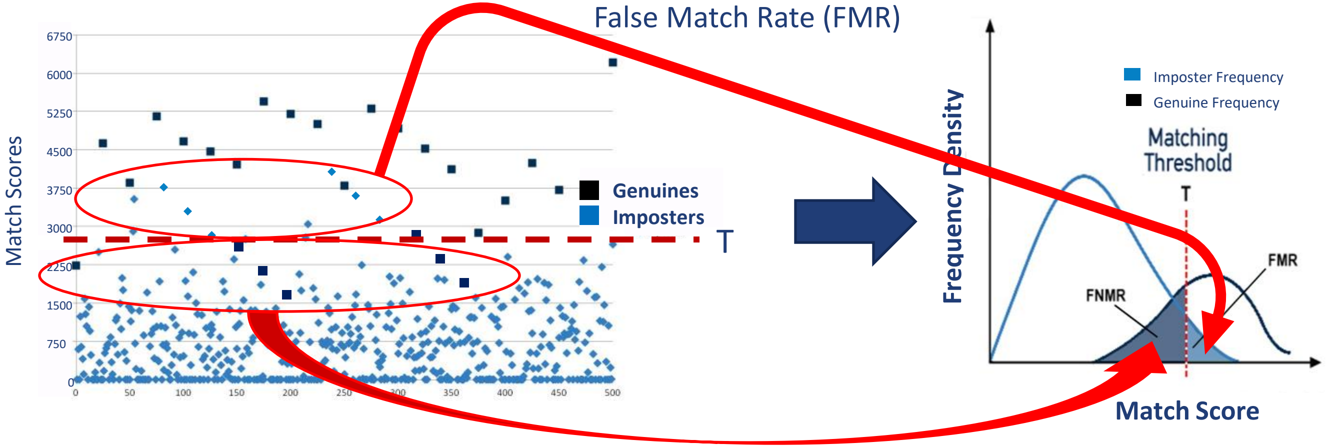
Introduction to Biometric Performance Metrics





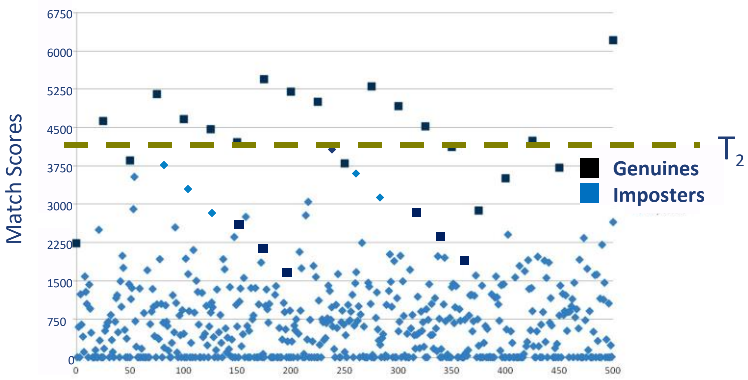
Terminology: FNMR vs. FMR

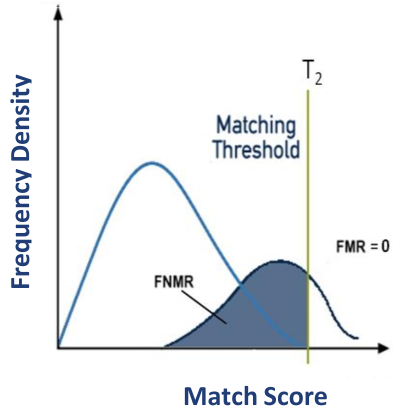
False Match Rate (FMR)



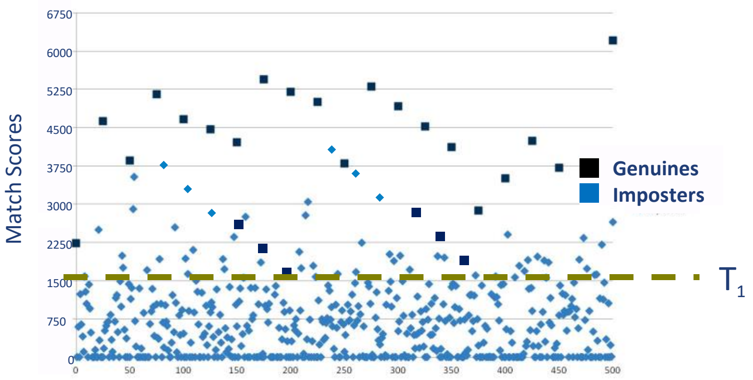
False Non-Match Rate (FNMR)

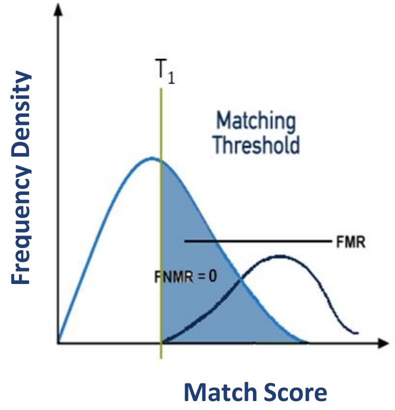
Terminology: FNMR vs. FMR



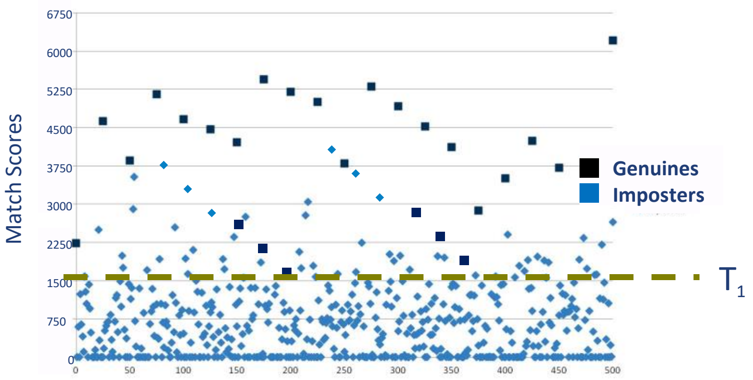


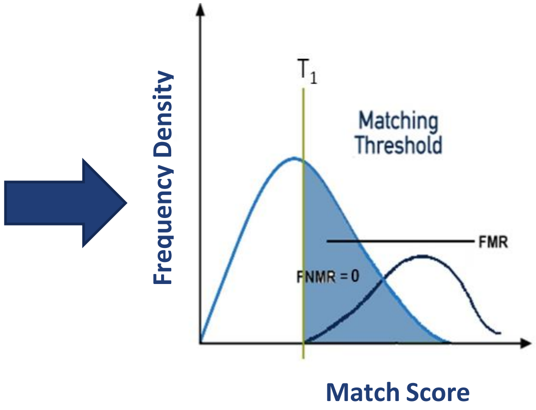
Terminology: FNMR vs. FMR

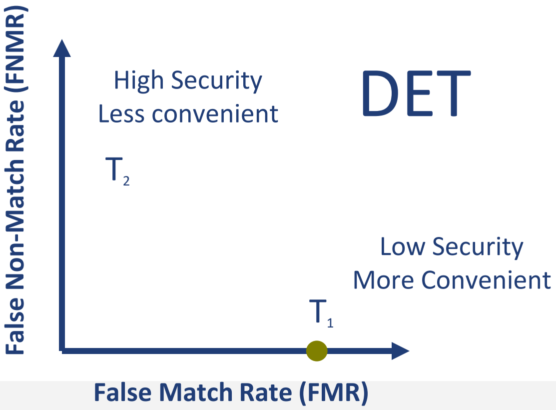




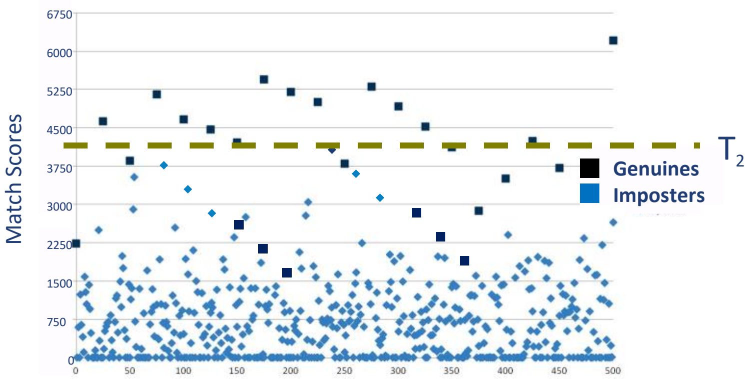
Terminology: FNMR vs. FMR

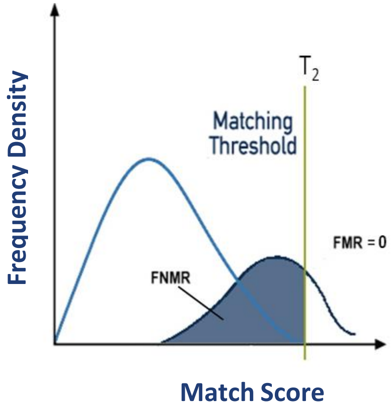


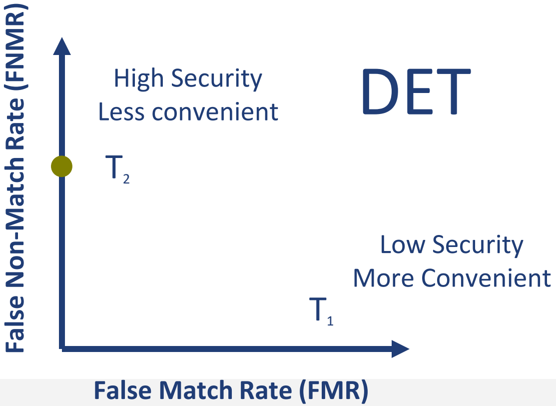




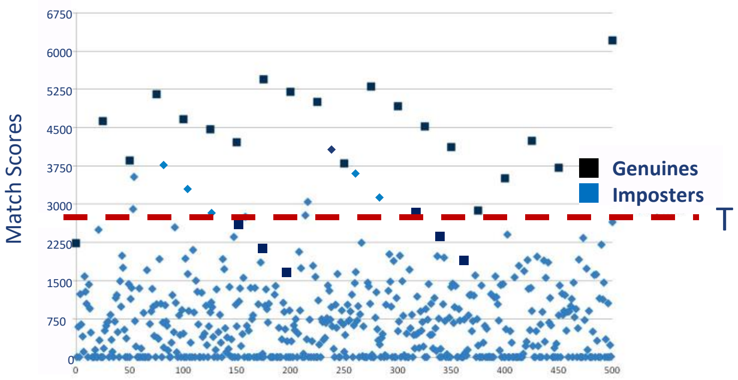
Terminology: FNMR vs. FMR

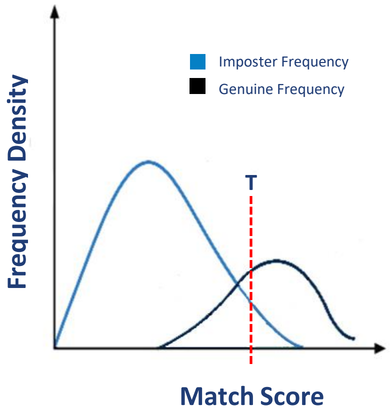


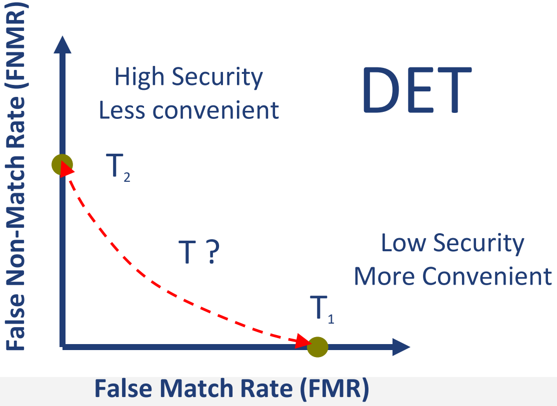




Terminology: FNMR vs. FMR

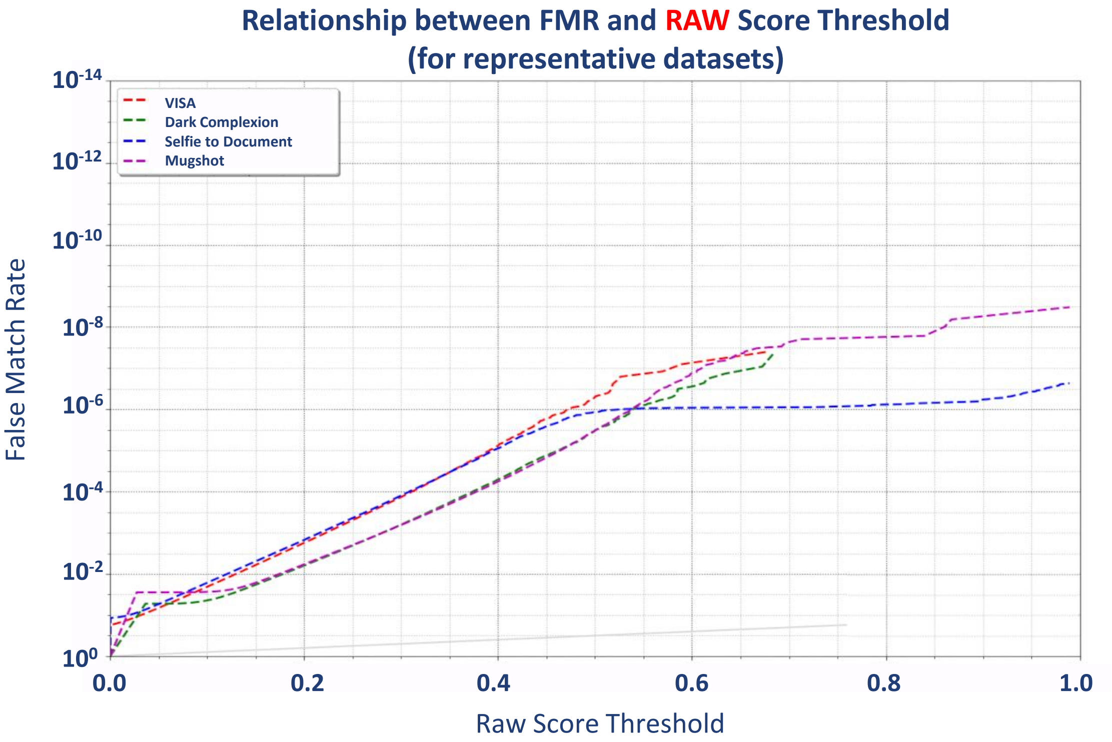






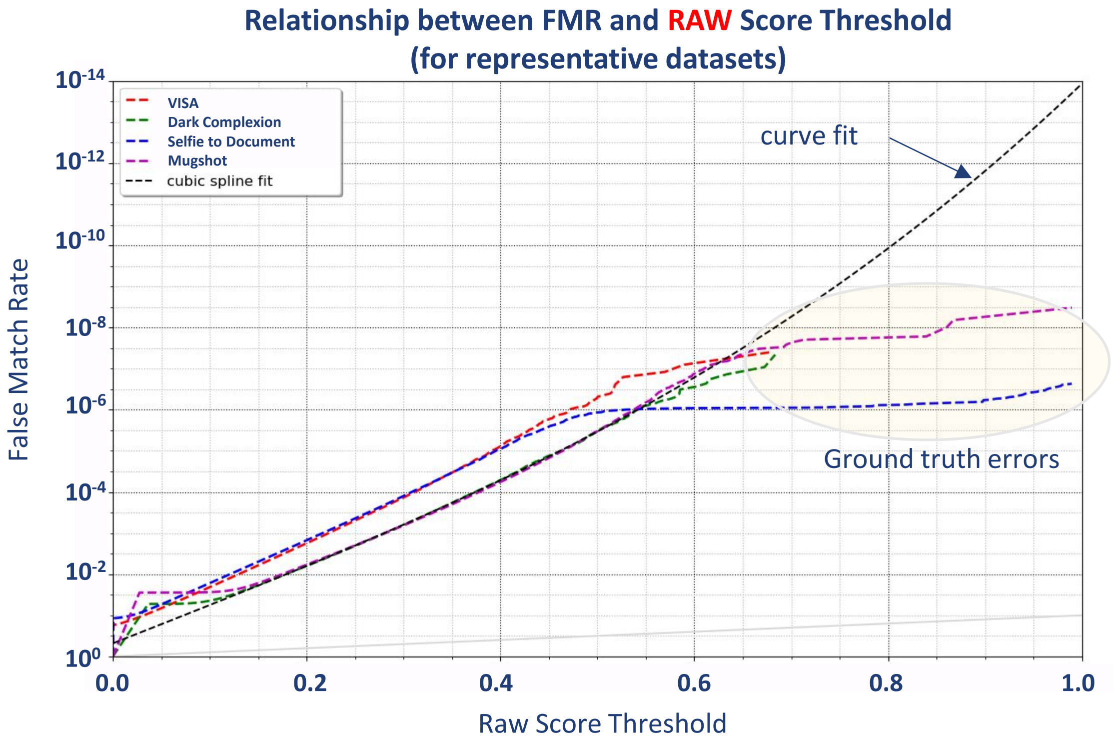
- relationship between matching algorithm thresholds and FMR is very stable
- function between FMR and threshold can be empirically determined
- use FMR instead of threshold!

Mapping Thresholds to FMR



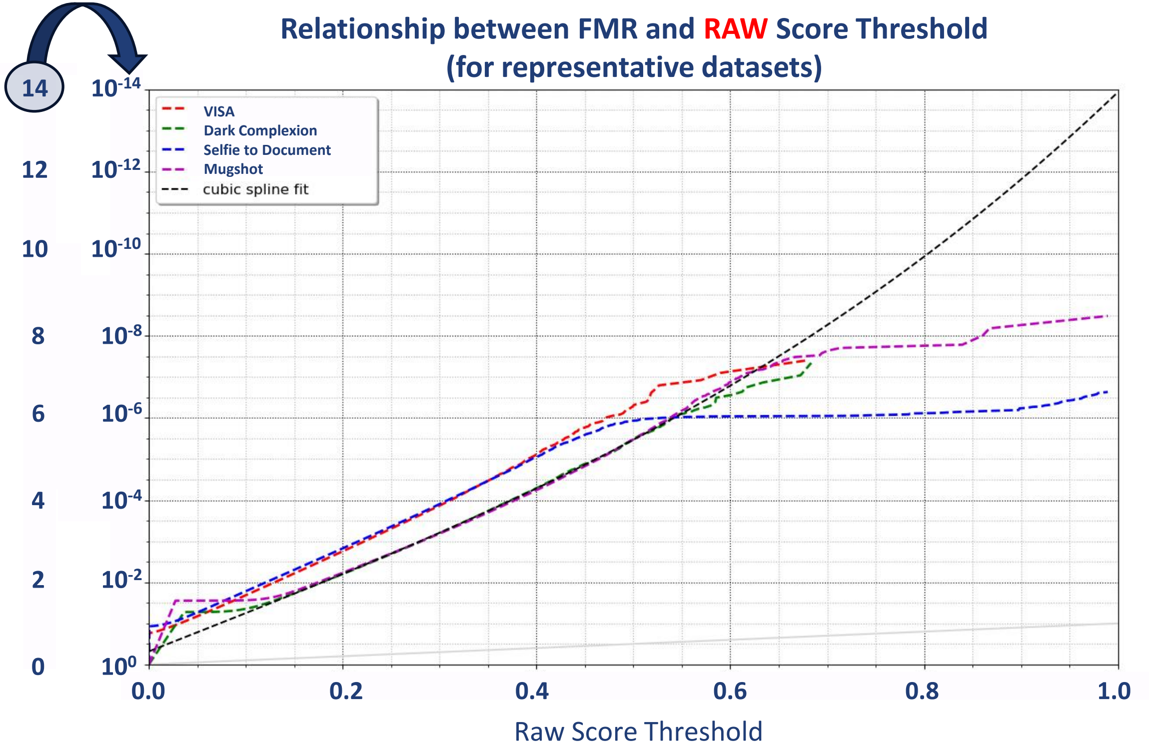
- Plot score threshold vs. FMR for various representative datasets

Mapping Thresholds to FMR



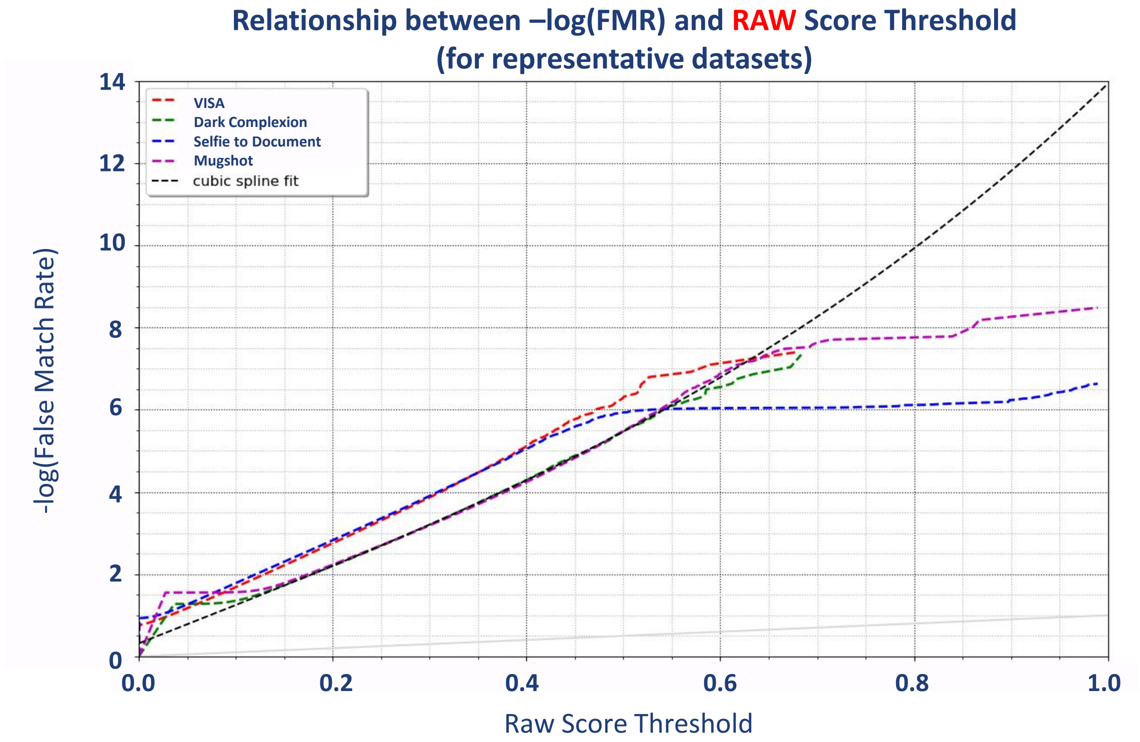
- Plot score threshold vs. FMR for various representative datasets
- Remove outliers and fit a curve
  - enables mapping of algorithm score into an FMR-based score
  - allows setting of thresholds according to desired FMR

Mapping Thresholds to FMR



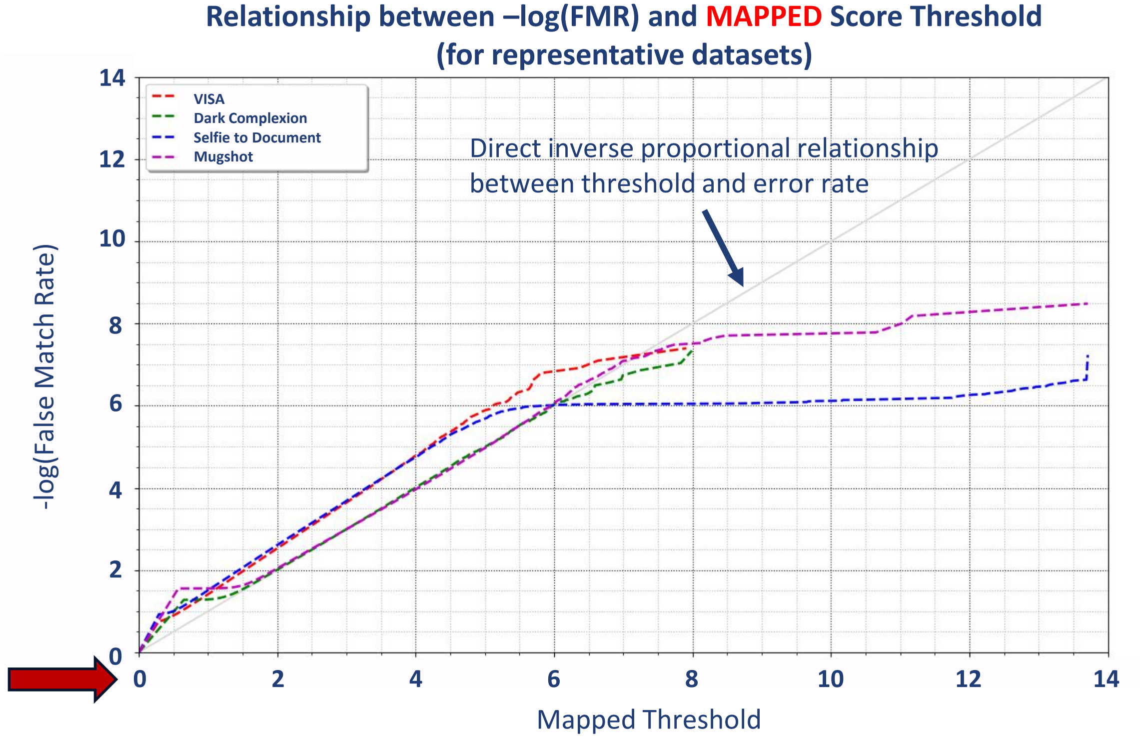
- Plot score threshold vs. FMR for various representative datasets
- Remove outliers and fit a curve
  - enables mapping of algorithm score into an FMR-based score
  - allows setting of thresholds according to desired FMR
- Simplify further by mapping to –log(FMR) instead!

Mapping Thresholds to –log(FMR)



- Plot score threshold vs. FMR for various representative datasets
- Remove outliers and fit a curve
  - enables mapping of algorithm score into an FMR-based score
  - allows setting of thresholds according to desired FMR
- Simplify further by mapping to –log(FMR) instead!

Final Mapped Thresholds to –log(FMR)



- Plot score threshold vs. FMR for various representative datasets
- Remove outliers and fit a curve
  - enables mapping of algorithm score into an FMR-based score
  - allows setting of thresholds according to desired FMR
- Simplify further by mapping to –log(FMR) instead!
- Monotonic mapping preserves ordering and does not affect DET

FMR – Based Thresholding

$$FMR = 10^{-T}$$

- intuitive relationship between operational threshold and an error rate relevant to the user
- enables consistent operational thresholds for FMR as accuracy and algorithms continue to improve

Example 1:

Set threshold to : 3

Expected System FMR = 10^{-3} **(1/1000)**

Example 2:

Set threshold to : 4

Expected System FMR = 10^{-4} **(1/10000)**

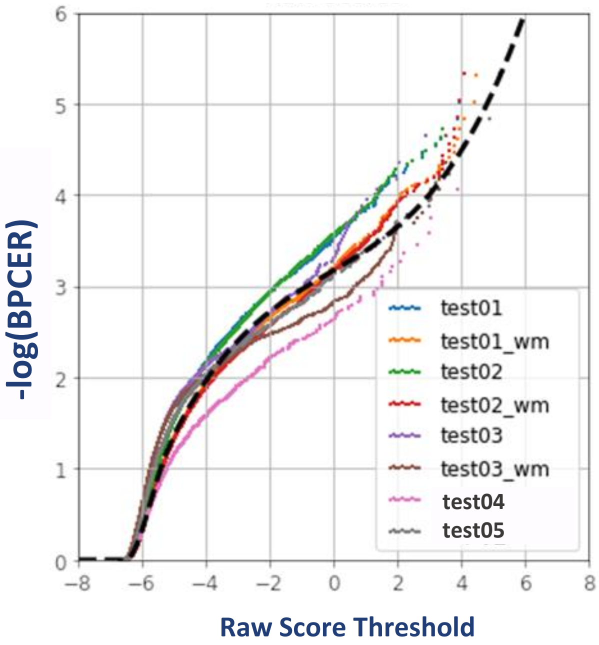
Example 3:

Set threshold to : 6

Expected System FMR = 10^{-6} **(1/1000000)**

-log(BPCER) Score Mapping

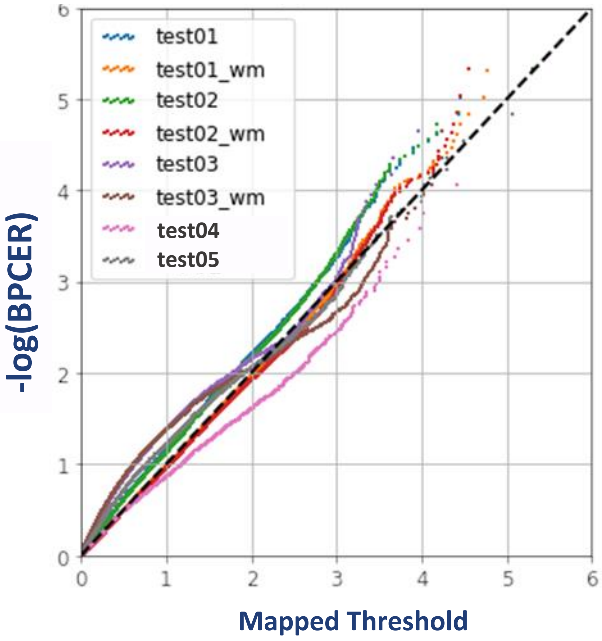
Relationship between –log(BPCER) and RAW Score Threshold (for representative datasets)



- only live data used for mapping
- analogous to the use of imposter data for matching

After Mapping

Relationship between –log(BPCER) and MAPPED Score Threshold (for representative datasets)



BPCER – Based Thresholding

$$BPCER = 10^{-T}$$

- intuitive relationship between operational threshold and an error rate relevant to the user
- enables consistent operational thresholds for BPCER as accuracy and algorithms continue to improve

Example 1:

Set threshold to : 1.0

Expected System BPCER = 10^{-1} **(1/10 expected live errors)**

Example 2:

Set threshold to : 2.0

Expected System BPCER = 10^{-2} **(1/100 expected live errors)**

Example 3:

Set threshold to : 3.0

Expected System BPCER = 10^{-3} **(1/1000 expected live errors)**

Science Forward: The Intuition Behind BPCER-based Score Mapping

- Modeling of live data seems to be more stable than modeling spoof data
  - ➢ Feature sets required to detect diversity of spoof species are likely more diverse than those required for live data – more challenging for models to learn score consistency?
  - ➢ Spoofs are open-ended with a constant evolution of attack vectors
- Abundancy of live data given expected normal operation enables more accurate modeling of score distributions with respect to BPCER
- Security is important, but usability seems to be a consistent concern operationally, so accurate assessment of its potential impact on the entire system is vital
- Why not APCER-based score mapping?
  - ➢ Given observed differences in spoof species detection error rates, APCER mapping would be sensitive to balance of spoof species in training vs. operational scenarios
  - ➢ Same security settings for different spoofs would incur the most usability error for the least accurate algorithm, possibly one that might be least prevalent

AWARE

Thank you!

Terry Riopka

triopka@aware.com