Proposed Addition of the XX.996 Hash Field

Patrick Grother

Information Technology Laboratory
National Institute of Standards and Technology (US),
United States Department of Commerce
What is this? As drafted ...

» Compute cryptographic hash over image data in XX.999
   » XX is Type 10, 13, 14, 15, 16, 17, 18, 19, 20, 99
   » XX is not used for Types 4 to 9

» Result is 64 characters
   » Hexadecimal [A-F,0-9] -- not base64

» It's a biometric template ... of sorts
   » Useful for rapid search for duplicated entries
   » Unique for any unique XX.999
   » If second sample has any difference → false non-match 😞
   » But template is not easy to reverse 😊
# Hash Field :: Why? And Why Not?

## PROs

- If the field is set for all images in a set, you can **detect byte-for-byte duplicates** (which do occur, operationally)
- Detection of bits being flipped during transmission (**channel errors**)
- Detection of clerical / unintended modifications, e.g. someone modifying the image and forgetting to update the hash.

## CONs

- It’s not a digital signature, so offers zero protection against a substitution attacks.
- For the byte-for-byte de-duplication task, it can always be computed on the ABIS / server side.
- Will not find rescanned faces
- It takes about 25 milliseconds per megabyte of data.
- Transmission time for 64 ASCII chars
So, what to do?

/> Reject
   » Insufficient value

/> Accept with modifications
   » Use “md5sum” instead of “sha256”
      • 32 bytes versus 64 bytes
      • Don’t need cryptographic strength
      • 18 milliseconds per megabyte (vs. 25).

/> Add it for
   • The face in Type 11, and SMT in Type 10.
   • Type 9?

/> Change type for Numeric “N” to Alphanumeric “AN”
Thank You
patrick.grother@nist.gov