Fusion of Face Recognition Algorithms (FOFRA) Prize Challenge 2018: Overview

Fusion To Improve Recognition Accuracy

Patrick Grother, Mei Ngan, NIST
Chris Boehnen, IARPA
Lars Ericson, IARPA (ctr)

2018-04-16

Fusion of Face Recognition Algorithms Prize Challenge 2018: Better Face Recognition via Fusion

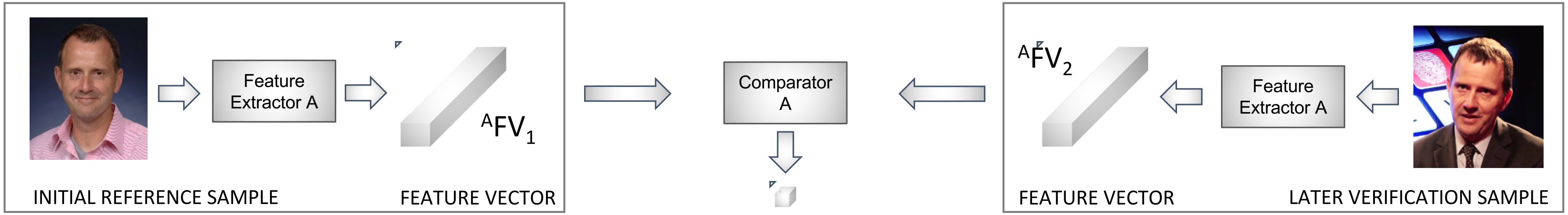
- There is a large literature on biometric fusion intended to improve accuracy via fusion of multiple modalities (face + fingerprint), multiple algorithms (ASM + CNN), or multiple samples (either contemporaneous or longitudinal).
- Most of the research has addressed 1:1 verification at the score-level.
- FOFRA 2018 is aimed at improved 1:N identification accuracy via template-level fusion
- Additional aims are: Template fusion for 1:1, and for score level fusion for 1:1 and 1:N

1:1 Verification in Operation

Three background slides on

- 1:1 verification without fusion
- 1:1 verification with score-level fusion
- 1:1 verification with template-level fusion

Basics 1: Traditional verification, no fusion



Feature Extractor A

$^{A}FV_1$

INITIAL REFERENCE SAMPLE

FEATURE VECTOR

Comparator A

$^{A}FV_2$

Feature Extractor A

FEATURE VECTOR

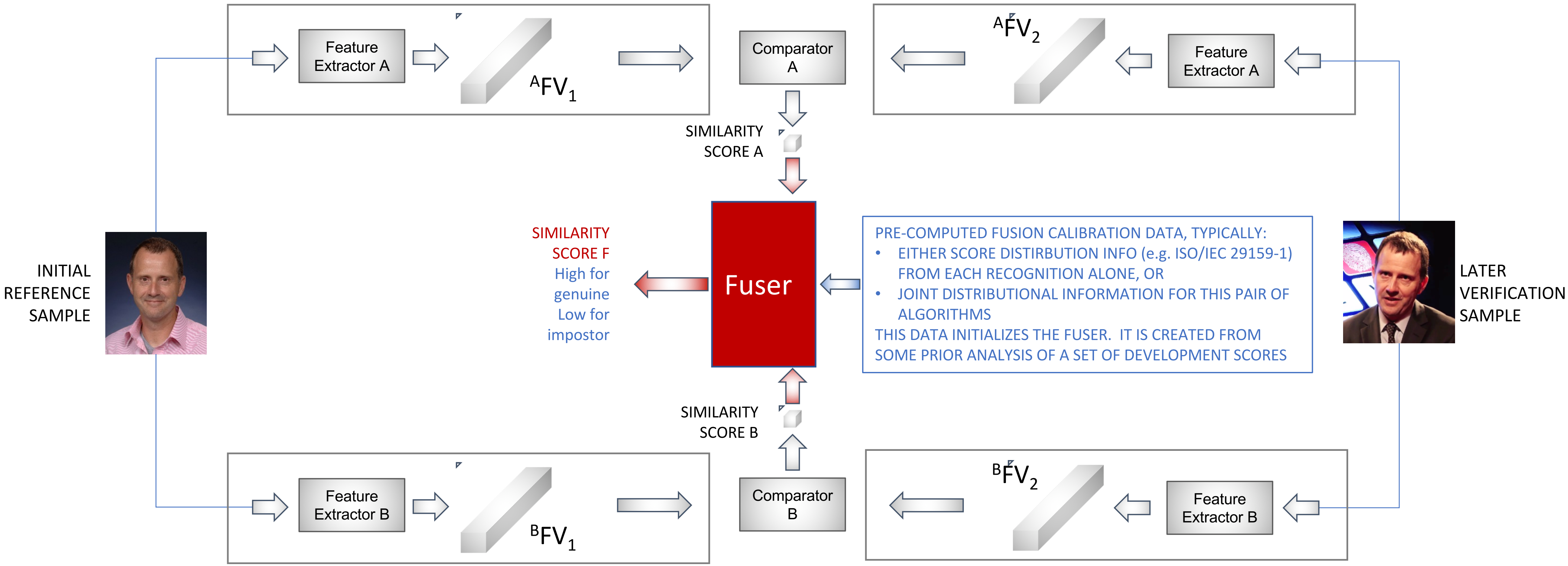
LATER VERIFICATION SAMPLE

SIMILARITY SCORE (SCALAR)

High for genuine

Low for impostor

Basics 2: Score fusion for multi-algorithm verification



INITIAL REFERENCE SAMPLE

Feature Extractor A

${}^{A}FV_1$

Comparator A

SIMILARITY SCORE A

${}^{A}FV_2$

Feature Extractor A

SIMILARITY SCORE F

High for genuine

Low for impostor

Fuser

PRE-COMPUTED FUSION CALIBRATION DATA, TYPICALLY:

- EITHER SCORE DISTRIBUTION INFO (e.g. ISO/IEC 29159-1) FROM EACH RECOGNITION ALONE, OR
- JOINT DISTRIBUTIONAL INFORMATION FOR THIS PAIR OF ALGORITHMS

THIS DATA INITIALIZES THE FUSER. IT IS CREATED FROM SOME PRIOR ANALYSIS OF A SET OF DEVELOPMENT SCORES

LATER VERIFICATION SAMPLE

SIMILARITY SCORE B

Feature Extractor B

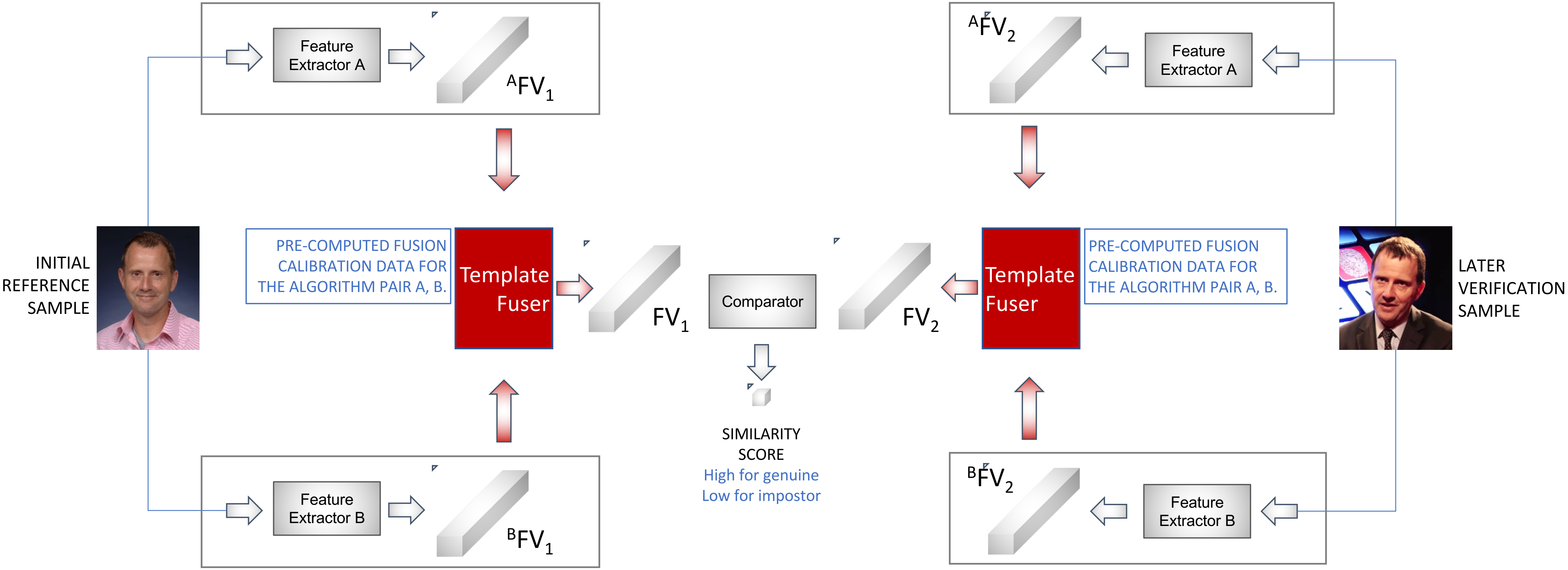
${}^{B}FV_1$

Comparator B

${}^{B}FV_2$

Feature Extractor B

Basics 3: Template fusion for multi-algorithm verification



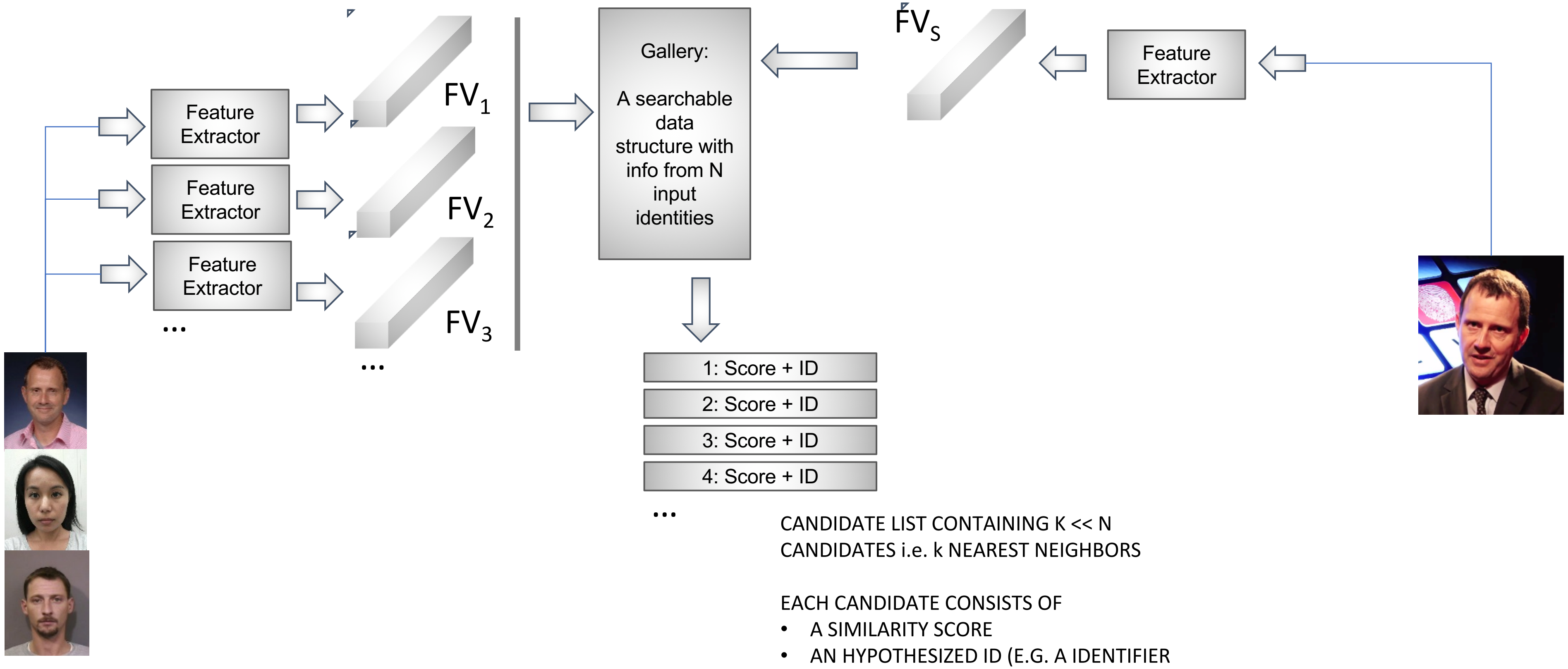
- Template = Feature vector + header
- Dimensions of A and B feature vectors will generally be different
- Assumption is template fusion will yield better recognition accuracy than score fusion
- A researcher must develop fusion scheme F(^{A}FV, ^{B}FV) and also a comparator, M(FV$_1$, FV$_2$)

1:N Identification in Operation

Three background slides on

- 1:N identification without fusion
- 1:1 identification with score-level fusion
- 1:1 identification with template-level fusion

Basics 2: Traditional identification, no fusion



INITIAL
REFERENCE
SAMPLES
AKA THE
GALLERY

…

N identities

Feature Extractor

Feature Extractor

Feature Extractor

…

FV_1

FV_2

FV_3

…

Gallery:

A searchable data structure with info from N input identities

1: Score + ID

2: Score + ID

3: Score + ID

4: Score + ID

…

FV_S

Feature Extractor

LATER SEARCH
SAMPLE

MAY OR MAY
NOT HAVE
MATE IN
AGLLERY

CANDIDATE LIST CONTAINING K << N
CANDIDATES i.e. k NEAREST NEIGHBORS

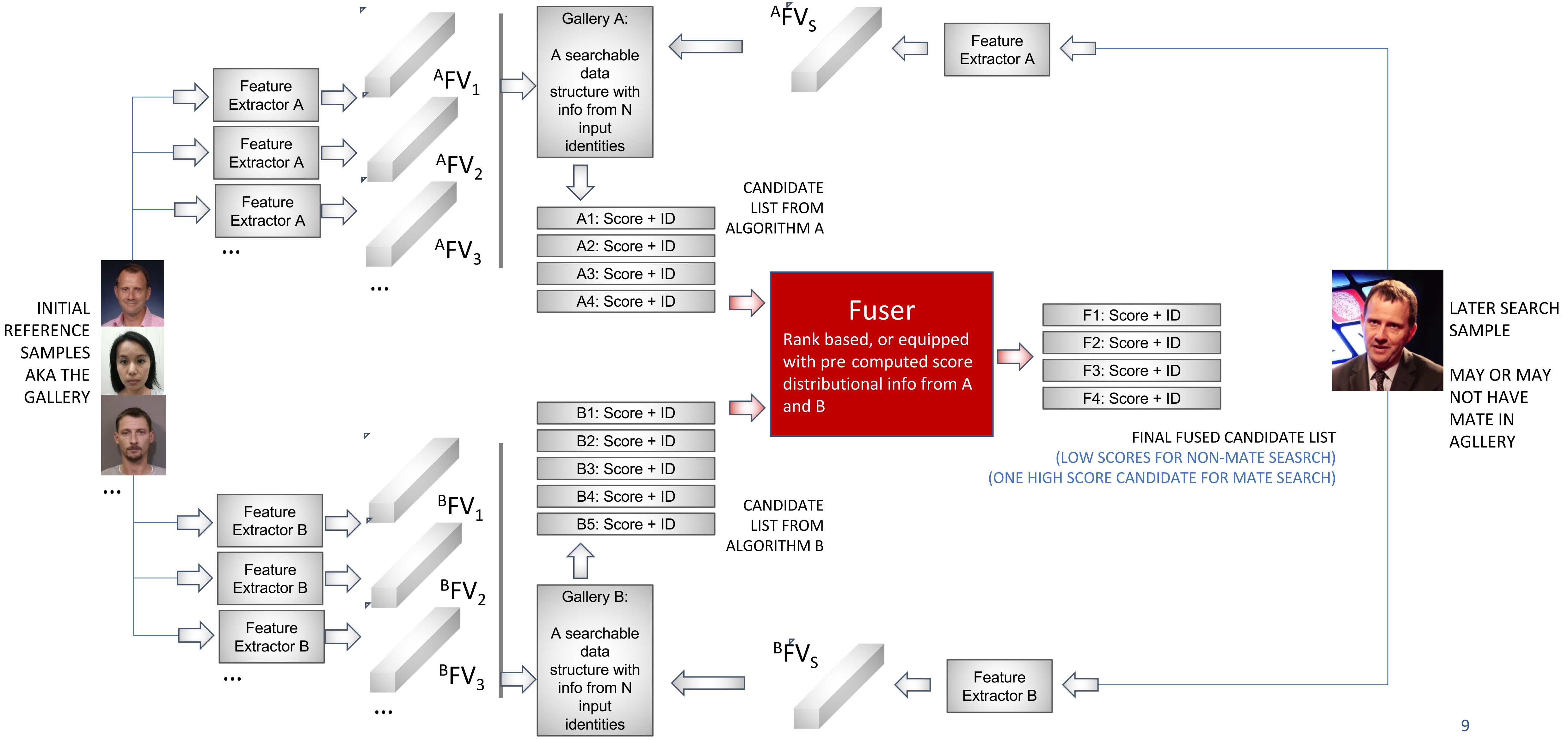
EACH CANDIDATE CONSISTS OF

- A SIMILARITY SCORE
- AN HYPOTHESIZED ID (E.G. A IDENTIFIER OR AN INDEX IN THE GALLERY)

NOTE

- THE CANDIDATE LIST IS SORTED IN DESCENDING ORDER OF SIMILARITY
- THE SAME ID CANNOT APPEAR ON THE CANDIDATE LIST TWICE

Basics 4: Score fusion for multi-algorithm identification



INITIAL REFERENCE SAMPLES AKA THE GALLERY

Feature Extractor A

Feature Extractor A

Feature Extractor A

…

$^{A}FV_{1}$

$^{A}FV_{2}$

$^{A}FV_{3}$

…

Gallery A:

A searchable data structure with info from N input identities

A1: Score + ID

A2: Score + ID

A3: Score + ID

A4: Score + ID

CANDIDATE LIST FROM ALGORITHM A

$^{A}FV_{S}$

Feature Extractor A

Feature Extractor B

Feature Extractor B

Feature Extractor B

…

$^{B}FV_{1}$

$^{B}FV_{2}$

$^{B}FV_{3}$

…

Gallery B:

A searchable data structure with info from N input identities

B1: Score + ID

B2: Score + ID

B3: Score + ID

B4: Score + ID

B5: Score + ID

CANDIDATE LIST FROM ALGORITHM B

$^{B}FV_{S}$

Feature Extractor B

Fuser

Rank based, or equipped with pre computed score distributional info from A and B

F1: Score + ID

F2: Score + ID

F3: Score + ID

F4: Score + ID

FINAL FUSED CANDIDATE LIST

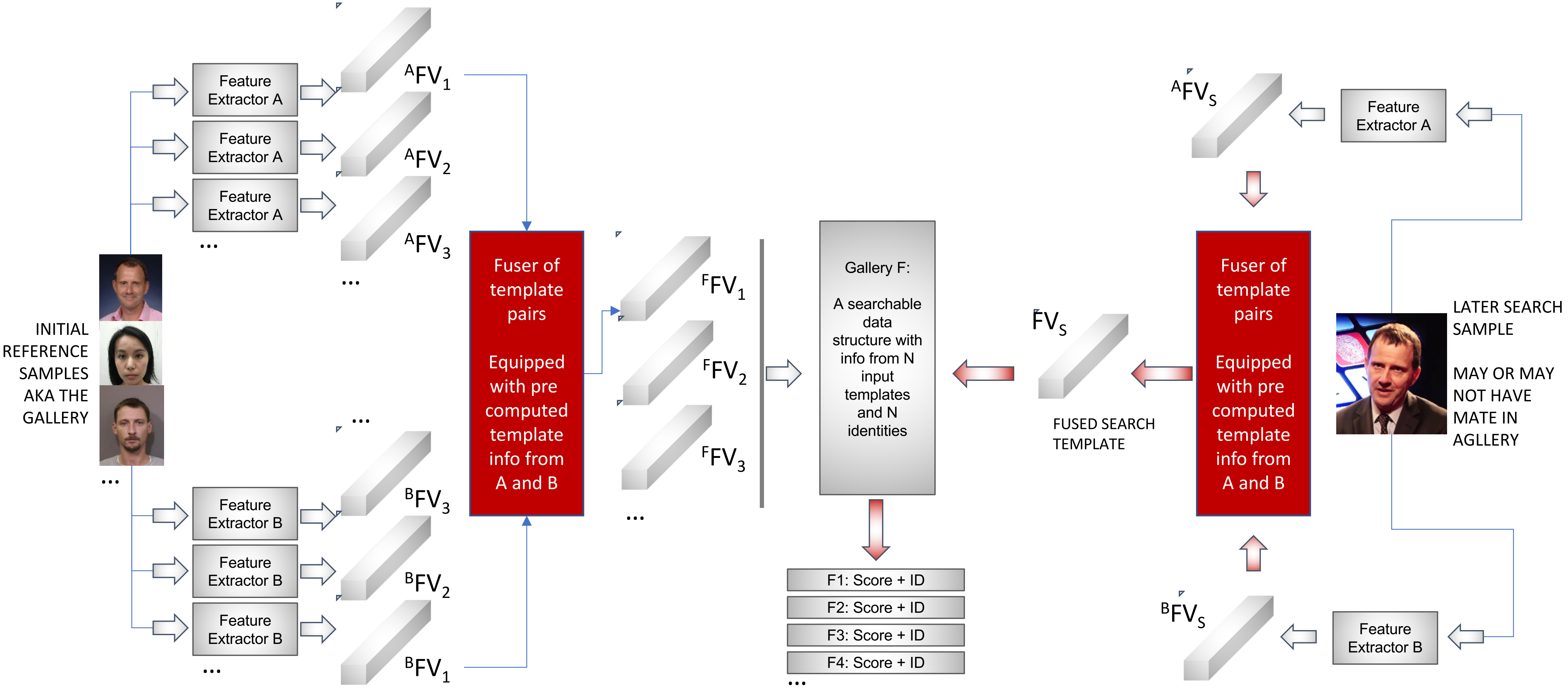
(LOW SCORES FOR NON-MATE SEASRCH)

(ONE HIGH SCORE CANDIDATE FOR MATE SEARCH)

LATER SEARCH SAMPLE

MAY OR MAY NOT HAVE MATE IN AGLLERY

Basics 4: Template fusion for multi-algorithm identification



INITIAL REFERENCE SAMPLES AKA THE GALLERY

Feature Extractor A

Feature Extractor A

Feature Extractor A

…

$^{A}FV_1$

$^{A}FV_2$

$^{A}FV_3$

…

Feature Extractor B

Feature Extractor B

Feature Extractor B

…

$^{B}FV_3$

$^{B}FV_2$

$^{B}FV_1$

…

Fuser of template pairs

Equipped with pre computed template info from A and B

$^{F}FV_1$

$^{F}FV_2$

$^{F}FV_3$

…

Gallery F:

A searchable data structure with info from N input templates and N identities

F1: Score + ID

F2: Score + ID

F3: Score + ID

F4: Score + ID

…

$^{F}FV_S$

FUSED SEARCH TEMPLATE

Fuser of template pairs

Equipped with pre computed template info from A and B

$^{A}FV_S$

Feature Extractor A

$^{B}FV_S$

Feature Extractor B

LATER SEARCH SAMPLE

MAY OR MAY NOT HAVE MATE IN AGLLERY

FINAL CANDIDATE LIST, LENGTH K << N

(LOW SCORES FOR NON-MATE SEASRCH)

(ONE HIGH SCORE CANDIDATE FOR MATE SEARCH)

The Fusion Challenges

Challenge #1: Score fusion

1. NIST provides set of scores for development
2. Developer submits fusion implementation
3. NIST evaluates on scores from same recognition algorithms applied on new sequestered image set
4. Prizes awarded on accuracy gains

Challenge #2: Template fusion

1. NIST provides set of templates for development
2. Developer submits fusion implementation
3. NIST evaluates on templates from same recognition algorithms applied to on new sequestered image set
4. Prizes awarded on accuracy gains

Challenge #1: Score Fusion

1. **Verification:** Fuse scores from 1:1 comparisons
2. **Identification:** Fuse candidate lists from 1:N searches

Score development set: Provided by NIST to developers

Verification

N genuine scores from comparison of gallery and probe templates, with IDs

M >> N

M impostor scores from comparison of gallery and probe templates, with IDs

Recognition Algorithm A

Scores from identical images, processed by second algorithm

Algorithm B

Scores from identical images, processed by third algorithm

Algorithm C

…

Number of algorithms is TBD

- Format
  - Text, XML, R (to be determined)
- Source
  - NIST Image set
- Subject IDs
  - Arbitrary integer labels, NIST assigned
- Score quantities
  - Verification: M ~ 10,000,000, N ~ 100,000

Identification

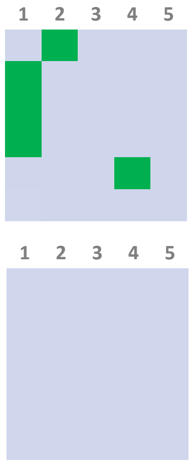
Candidate lists will have length 20, but here length is 5. Six candidate lists are shown. Green denotes position of mate.

Some mated searches fail to yield the mate (last row)

Non-mated searches only produce impostor scores, in grey.

J candidate lists scores from mated searches of a gallery. Each row is a candidate list.

M candidate lists from non-mated searches of a gallery



Recognition Algorithm A

Candidate lists from identical images, processed by second algorithm

Algorithm B

Candidate lists from identical images, processed by third algorithm

Algorithm C

…

Number of algorithms is TBD

- Score quantities
  - Identification: J ~ 100,000, M ~ 100,000

Participant deliverables to NIST: Fusion and recognition functions

Identification apparatus:

1. Fuser for a candidate list from A and a candidate list from B.

Verification apparatus:

1. Fuser of a score from A with a score from B

Participant X for A-B output scores

Similarly:
Participant X for A-C output scores

Similarly for three input scores:
Participant X for A-B-C output scores

...

i.e. for each algorithm pair, developers should submit functions that fuse scores from 1:1 verification, and fuse candidate lists from 1:N searches

- Developer elects to submit open-source or closed.
- Format
  - Open-source: R code, or C++ source code
  - Closed-source: Compiled library implementing NIST-specified C++ API
- Coverage:
  - All pairs of provided algorithms
  - Three-way score fusion (e.g. XYZ) will not count toward prize award
- Audience
  - Worldwide companies or universities with interest and capability to do fusion.

NIST evaluation of score fusion schemes

1. NIST establishes test images and algorithms

- 1:1
- 1:N open set

2. NIST runs recognition trials

- Produce 1:1 comparison scores from recognition algorithms X, Y, Z
- Produce 1:N candidate lists from from recognition algorithms X, Y, Z
- Execute developer's 1:1 score fusion function
  - for XY, for XZ etc.
- Execute developer's 1:N candidate list fusion function
  - for XY, for XZ etc.

3. NIST computes accuracy figure of merit

- Verification: FNMR at FMR = 0.0001
- Identification: FNIR at FPIR = 0.003, N ~ 1million
- Compute these for each input algorithm alone (X, Y, etc)
- Compute these for each developers fusion
  - Developer 1 fusing XY; Developer 2 fusing XZ etc.

FNIR: Proportion of mated searches for which mate is not returned at or above a threshold T
FPIR: Proportion of non-mated searches yielding one or more candidates above same threshold

4. NIST computes accuracy gains

- For each developer
  - For pair AB
    - Compute reduction in recognition error rates achieved using fusion over the native accuracy
    - Verification: $\text{FNMR}_{\text{FUSED}} - \min(\text{FNMR}^{X}_{\text{NATIVE,}}\ \text{FNMR}^{Y}_{\text{NATIVE}})$
    - Identification: $\text{FNIR}_{\text{FUSED}} - \min(\text{FNIR}^{X}_{\text{NATIVE,}}\ \text{FNIR}^{Y}_{\text{NATIVE}})$
  - Repeat for XZ ...

5. NIST ranks developers

- Rank the developers by computing the best mean error rate reductions, taking the mean over all pairs XY, YZ etc.

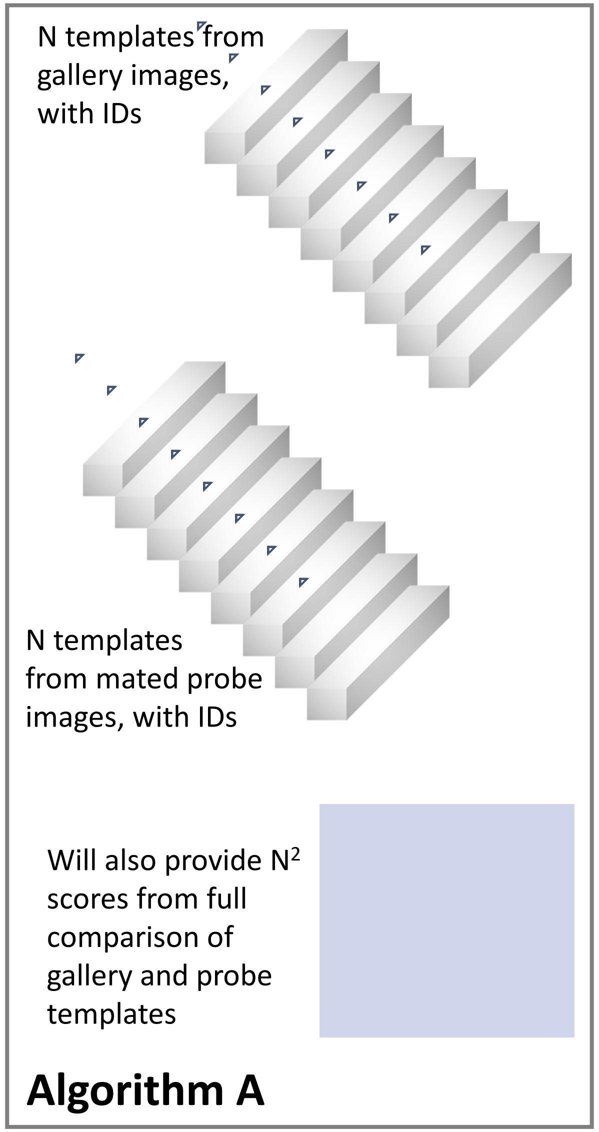
6. NIST reports to IARPA appointed judges, who

- Consider NIST results report
- Adjudicate ties, weigh software reliability, speed, conformance to specification, data irregularities, unexpected effects (e.g. fusion doesn't work) etc
- Award prizes to best error rate reduction

Challenge #2: Template Fusion

1. Verification:
   - Fuse reference templates
   - Fuse probe templates, then compare (1:1) with reference
2. Identification:
   - Fuse N pairs of templates, and build gallery
   - Fuse probe templates, then search (1:N) gallery

Template development set: Provided by NIST to developers



N templates from gallery images, with IDs

N templates from mated probe images, with IDs

Will also provide N^2 scores from full comparison of gallery and probe templates

Algorithm A

Templates and scores from identical images, processed by second algorithm

Algorithm B

Templates and scores from identical images, processed by third algorithm

Algorithm C

...

Number of algorithms is TBD

- Format
  - Text, XML, R (to be determined)
- Quantity
  - N ~ 100,000
- Source
  - NIST image set
- Subject IDs
  - Arbitrary integer labels, NIST assigned
- Templates
  - Will be real-valued feature vectors, derived from NIST image set

Participant deliverables to NIST: Fusion and recognition functions

Identification apparatus:

1. Gallery constructor for A templates
2. Search function for A templates against gallery

Verification apparatus:

1. Comparator function for A templates

Participant X for A templates alone

As in box above

Participant X for B templates alone

…

i.e. for each set of templates, developers should submit 1:1 and 1:N functions for matching un-fused "as-is" templates.

Identification apparatus:

1. Template fuser for AB
2. Gallery constructor for AB templates
3. Search function for AB templates against gallery

Verification apparatus:

1. Template fuser for AB
2. Comparator function for AB templates

Participant X for A-B templates

As in box above

Participant X for A-C templates

As in box above

Participant X for A-B-C templates

…

i.e. for each algorithm combination, (pairs, triples …) developers should submit 1:1 and 1:N functions for making and recognizing fused templates

- Developer elects to submit open-source or closed.
- Format
  - Open-source: R code, or C++ source code
  - Closed-source: Compiled library implementing NIST-specified C++ API
- Coverage:
  - All combinations of provided algorithms
- Audience
  - Worldwide companies or universities with interest and capability to do fusion.

NIST evaluation of template fusion schemes

1. NIST establishes test images and algorithms

- 1:1
- 1:N open set

2. NIST runs recognition trials

- Make template from algorithm standalone X, Y, Z etc.
- Fuse them
- Execute developer’s verification function on fused templates
    - Fused XY, XZ, etc. and also XYZ etc.
- Execute developer’s identification functions on fused templates
    - Fused XY, XZ etc. and also XYZ etc.

3. NIST computes accuracy figure of merit

- Verification: FNMR at FMR = 0.0001
- Identification: FNIR at FPIR = 0.003, N ~ 1million

FNIR: Proportion of mated searches for which mate is not returned at or above a threshold T
FPIR: Proportion of non-mated searches yielding one or more candidates above same threshold

4. NIST computes accuracy gains

- For each developer
    - For pair XY
        - Rank developer implementations by reduction in error rates using fusion over the native accuracy
        - Verification: $\mathrm{FNMR}_{\mathrm{FUSED}} - \min(\mathrm{FNMR}^{X}_{\mathrm{NATIVE},}\ \mathrm{FNMR}^{Y}_{\mathrm{NATIVE}})$
        - Identification: $\mathrm{FNIR}_{\mathrm{FUSED}} - \min(\mathrm{FNIR}^{X}_{\mathrm{NATIVE},}\ \mathrm{FNIR}^{Y}_{\mathrm{NATIVE}})$
        - Native accuracy is computed using the recognition algorithm sequestered at NIST, not the matcher supplied for un-fused inputs.
    - Repeat for XZ ...

5. NIST ranks developers

- Two-Way Fusion: Rank the developers by computing the best mean error rate reductions, taking the mean over all pairs XY, YZ etc.
- Three-Way Fusion: Rank the developers by the best error rate reduction over the set of XYZ

6. NIST reports to IARPA appointed judges, who

- Consider NIST results report
- Adjudicate ties, weigh software reliability, speed, conformance to specification, data irregularities, unexpected effects (e.g. fusion doesn’t work) etc
- Award prizes to best reduction

Prizes

Prizes	Metric	Score level fusion	Template level fusion (Two-Way)	Template level fusion (Three-Way)
Verification	Reduction in FNMR	$2000 + open-source bonus $2000[+] + workshop attendance $4000[X] (max $8000)	$8000 + open-source bonus $4000[+] + workshop attendance $4000[X] (max $16000)	N/A
Identification	Reduction in FNIR	$5000 + open-source bonus $2000[+] + workshop attendance $4000[X] (max $11000)	$10000 + open-source bonus $4000[+] + workshop attendance $4000[X] (max $18000)	$9000 + open-source bonus $4000[+] + workshop attendance $4000[X] (max $17000)

[+] The extra prize is awarded if all of the following apply:

- The developer submits complete open-source software to NIST
- The developer gives written permission for NIST and IARPA to freely distribute the software
- The permission is provided to NIST at any time before NIST publishes the FOFRA 2018 results

[x] The extra prize is awarded if the participant sends representative who:

- Attends a meeting/conference where IARPA and NIST will present results
- Makes a 15 minute presentation on the fusion technology

Note

- A developer can only win one workshop attendance bonus ($5000), even if they win in multiple categories.
- This is being done in lieu of issuing invitational travel orders

Max purse: $70000

Min purse: $34000 if no attendance, no open source

Typical purse I: $44000 no open source, two developers win and attend workshop

Typical purse II: $34000 + 5000 one winner workshop + $16000 open-source = $55000