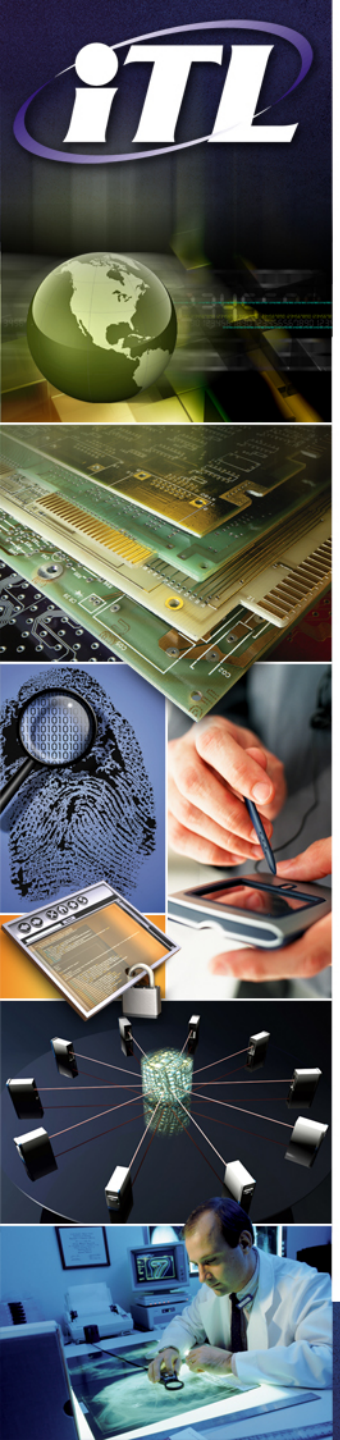


Six month PI meeting

MediFor Nimble Challenge Evaluation

Jan. 25, 2017

Jonathan Fiscus (PI)*
Dr. Haiying Guan (Presenter)*
Dr. Yooyoung Lee (Presenter)*
Dr. Amy Yates+
Andrew Delgado*
Daniel Zhou*
Timothee Kheyrkhah (GR)*

* Multimodal Information Group
+ Image Group
Information Access Division
Information Technology Laboratory
National Institute of Standards and Technology (NIST)

Thanks to the Test and Evaluation Team!

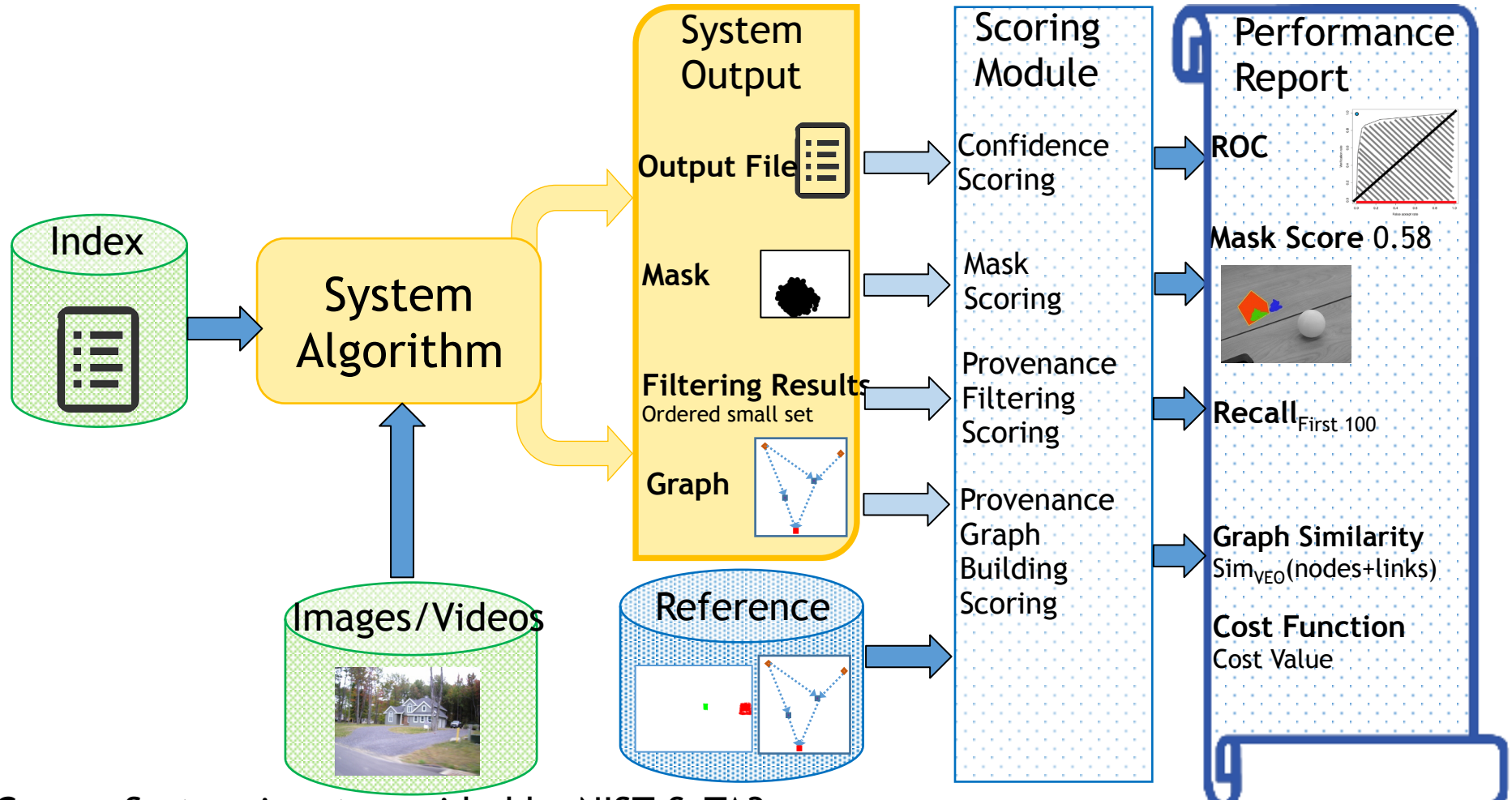
- DARPA Media Forensic (Medifor) Team – Role: Program administration
  - (http://www.darpa.mil/program/media-forensics)
- TA3 Team – Role: Data production and curation
  - PAR Government (http://www.pargovernment.com/)
  - National Center for Media Forensics, University of Colorado Denver (http://www.ucdenver.edu/academics/colleges/CAM/Centers/ncmf/Pages/ncmf.aspx)
  - RankOne (http://www.rankone.io/)
  - Rochester Institute of Technology
  - Drexel University
  - University of Michigan
- Air Force Research Lab – Role: Contracting
- NIST Medifor Team – Role: Evaluation designed and implementation

Outline

- NC2017 Tasks and Evaluation Schedule
- NC2017 Data
- Scoring Software: Detection and Localization
- Proposed Provenance Evaluation Metrics
- Open Issues/Discussion

NC2017 Tasks

Overview: Evaluation Modules & Data Flow



Green: System input provided by NIST & TA3
Yellow: Performer modules
Blue: NIST Evaluation modules

Nimble Challenge (NC) 2017 Tasks

- Manipulation Detection and Localization
  - Images: support selective manipulation scoring
  - Videos (detection only)
- Splice Detection and Localization
- Provenance Filtering
- Provenance Graph Building

Future Nimble Challenge Tasks

- Association
- Semantic Integrity

Manipulation Detection and Localization Task

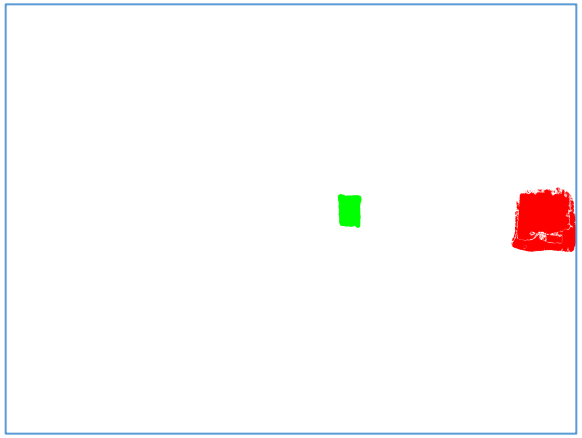
- Task Descriptions:
  - Detection: Given a probe image, detect if the probe was "manipulated".
  - Localization: If the probe is determined to be manipulated, indicate the region of the "localizable manipulations"
- Definitions:
  - "Manipulation" defined to be: splice, clone, remove, blur, laundering (media filter), anti-forensic …
  - "Localizable manipulations" are manipulations except global operations
- Input:
  - Image Task: Probe image
  - Video Task: Probe video (detection only)
- System input conditions:
  - Image/video Only (no header or metadata)
  - Image/video Only + camera fingerprint data
  - Image/video + Metadata
- Task Outputs:
  - Image: Confidence score and mask localization result (local manipulations)
  - Video: Confidence score

Manipulation Detection and Localization Example

Base Image
(original)

Probe Image
(manipulated)

Color Composite Mask



Manipulation Detection and Localization Evaluation Model

System Input

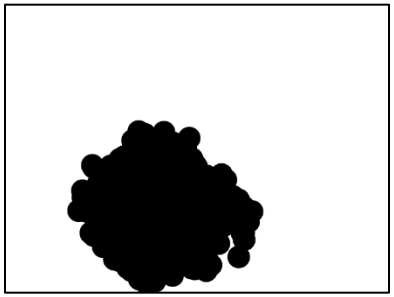
Image(s) +
(metadata)

Algorithm

System Output

Confidence score

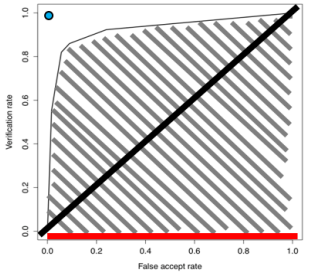
27.58



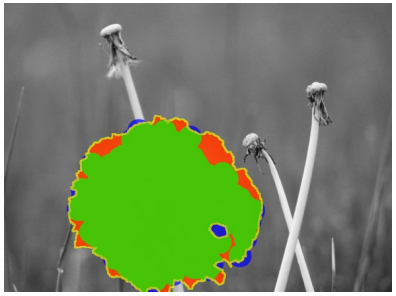
Probe Mask
(If a manipulation)

Metrics

Confidence Score
Receiver operating characteristic (ROC)



AUC: 0.5



Manipulated image
Mask Score: 0.58

Manipulation Detection and Localization Evaluation Model

System Input

Image(s) +
(metadata)

Algorithm

System Output

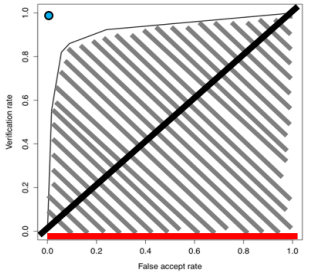
Confidence score

-17.58

Metrics

Confidence Score

Receiver operating characteristic (ROC)



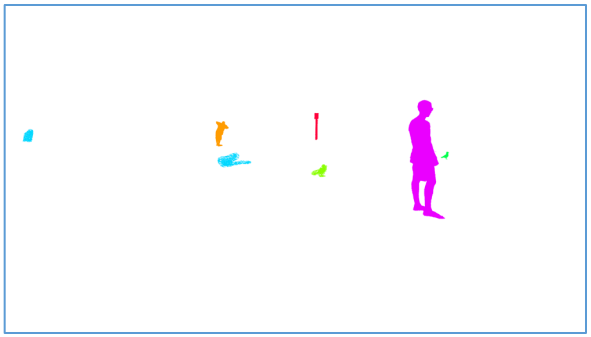
AUC: 0.5

Splice Detection and Localization Task

- The same as NC2016!
- Question: Was something from the donor spliced into the probe?
- Task Descriptions:
  - Detection: Given a probe image and a potential donor image, detect if something from the donor spliced into the probe
  - Localization: If a splice occurred, indicate the region splice in the probe and the donor respectively
- Task inputs
  - Probe image
  - A potential donor image
- System input conditions:
  - Image/video Only (no header or metadata)
  - Image/video + Metadata
- Task Outputs
  - Confidence score
  - Two Masks
    - Probe mask indicates where the spliced material was placed on the probe
    - Donor mask indicates where the spliced material was taken from the donor

Splice Detection and Localization Example

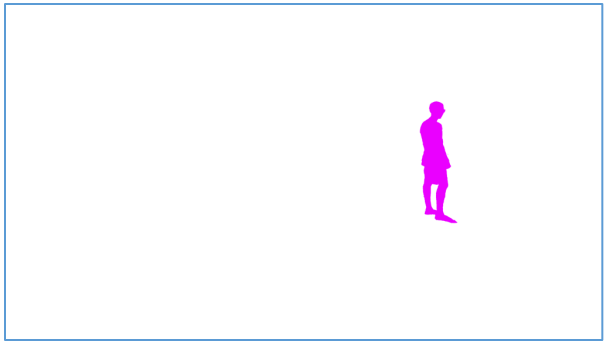
Color Composite Mask



Base Image

Probe Image

Reference Probe Mask Given the Donor



Donor Image

Donor Mask

Splice Detection and Localization Evaluation Model

System Input

Image(s) + (metadata)

Probe Image

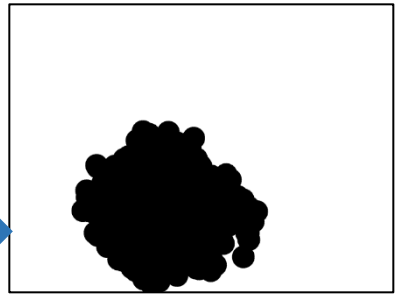
Donor image

Splice Detection Algorithm

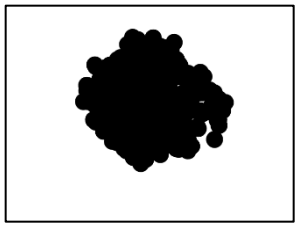
System Output

Confidence score

97.86



System output probe mask

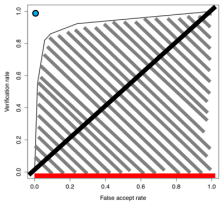


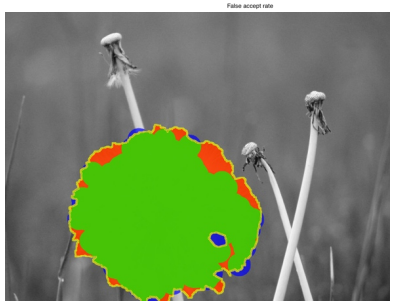
System output donor mask

Metrics

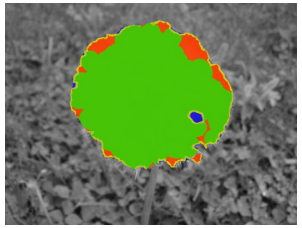
Confidence Score

ROC





Manipulated image Mask Score: Maximum Matthews Correlation Coefficient (MCC)



Donor image Mask Score: Maximum MCC

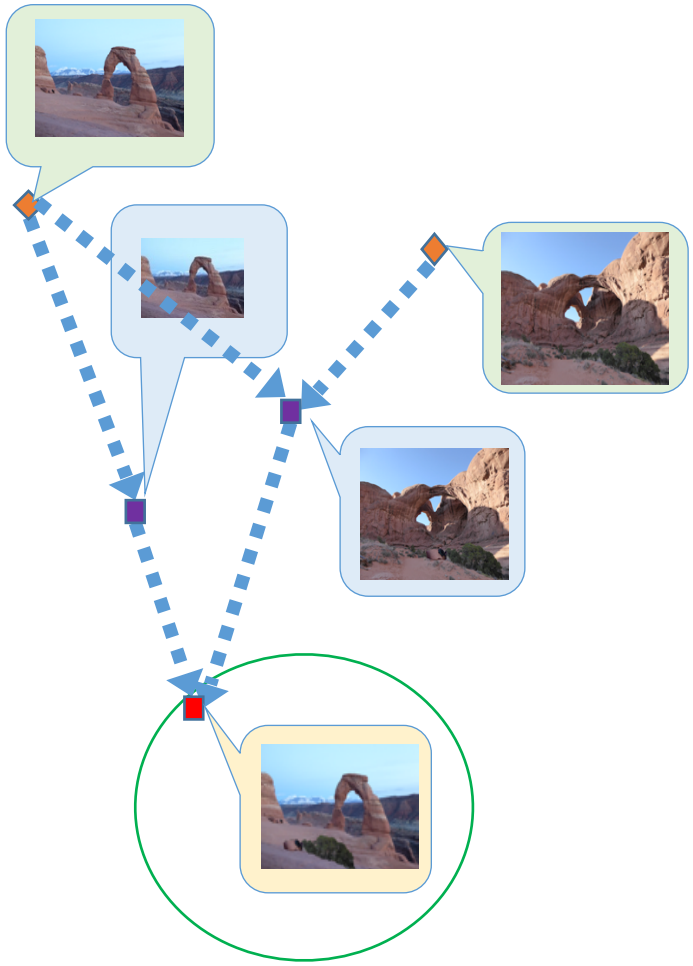
NC 2017 Provenance Tasks

The ultimate goal is to be able to generate and describe the links of a full provenance graph, from ANY node in the graph, a single donor, an intermediate node, and or modified leaf node.

Two tasks

- Provenance Filtering
- Provenance Graph Building

These tasks are viewed as a step toward the goal, but should not been seen as limiting future goals.



Provenance Filtering Task

- Task description
  - Given a probe image, return all images (its ancestors and descendants in the world dataset) in its genealogy graph.
- Task inputs
  - A probe image
  - A world dataset
- System input conditions:
  - Image/video Only (no header or metadata)
  - Image/video + Metadata
- Task outputs
  - For each probe, a set of N images as potential candidates with their confidence scores.

Provenance Filtering Example

Probe Image

NC2017 Evaluation World Set (≈1M)

Provenance Filtering Example

Probe Image

NC2017 Evaluation World Set (≈1M)

Provenance Filtering Example

Probe Image

NC2017 Evaluation World Set (≈1M)

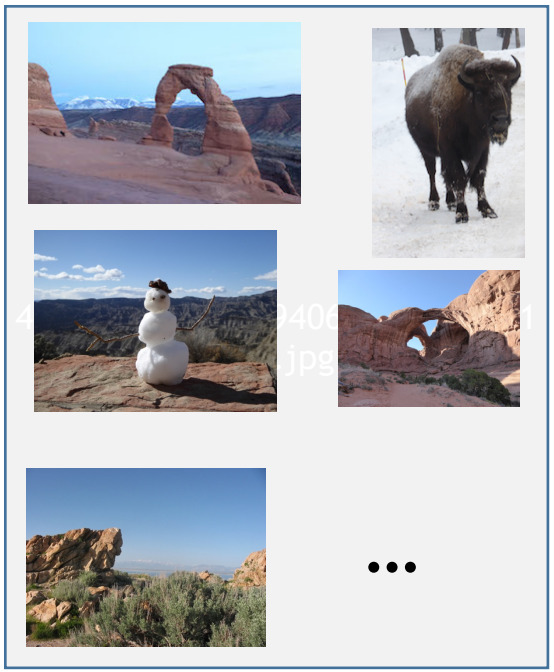
First 100 images' recall

Provenance Filtering Evaluation Model

System Input

Probe Image

World Image Set (≈1M)



Algorithm

System Output

A set of N images with confidence score

27.58

25.58

17.58

2.58

...

Metrics

Recall First 100

Recall First 50

Provenance Graph Building
(Task Formulation Underway)

- Task Description
  - Given a probe image, construct and label the manipulation provenance graph that includes all its ancestors and descendants in the world dataset.
- Task Inputs
  - End-to-End Provenance: a probe image, a large world set (1M images)
  - Oracle Provenance: a probe image, a small world set (≈100 images, all contributor images with some distractor world images)
- System input conditions:
  - Image Only
  - Image + Metadata
- Task outputs:
  - a provenance graph

Provenance Graph Building System Input

Probe Image

NC2017 Evaluation World Set





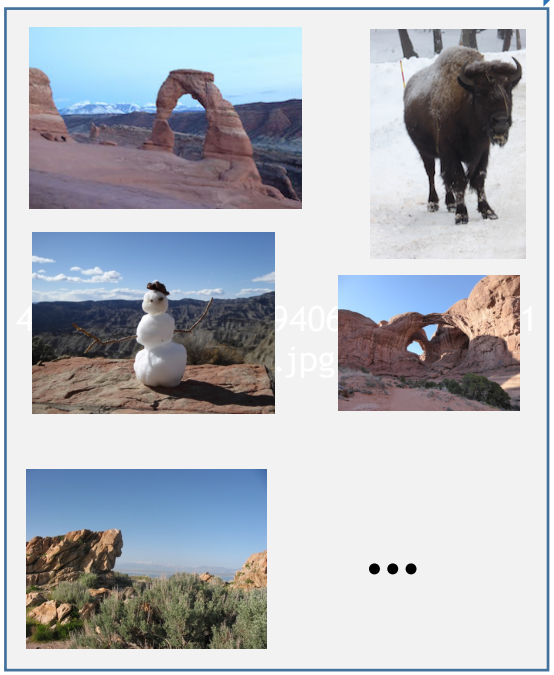


Provenance Graph Building Evaluation Model

System Input

Probe Image

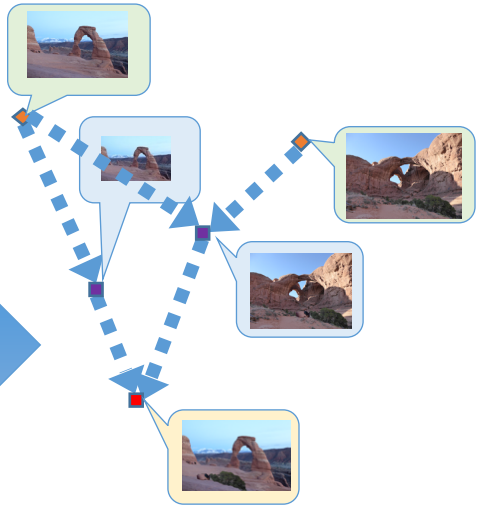
(1) World Image Set (≈1M)
(2) Oracle Set (≈100)



Algorithm

System Output

A provenance graph



Metrics

Graph Similarity

Generalized F-measure:

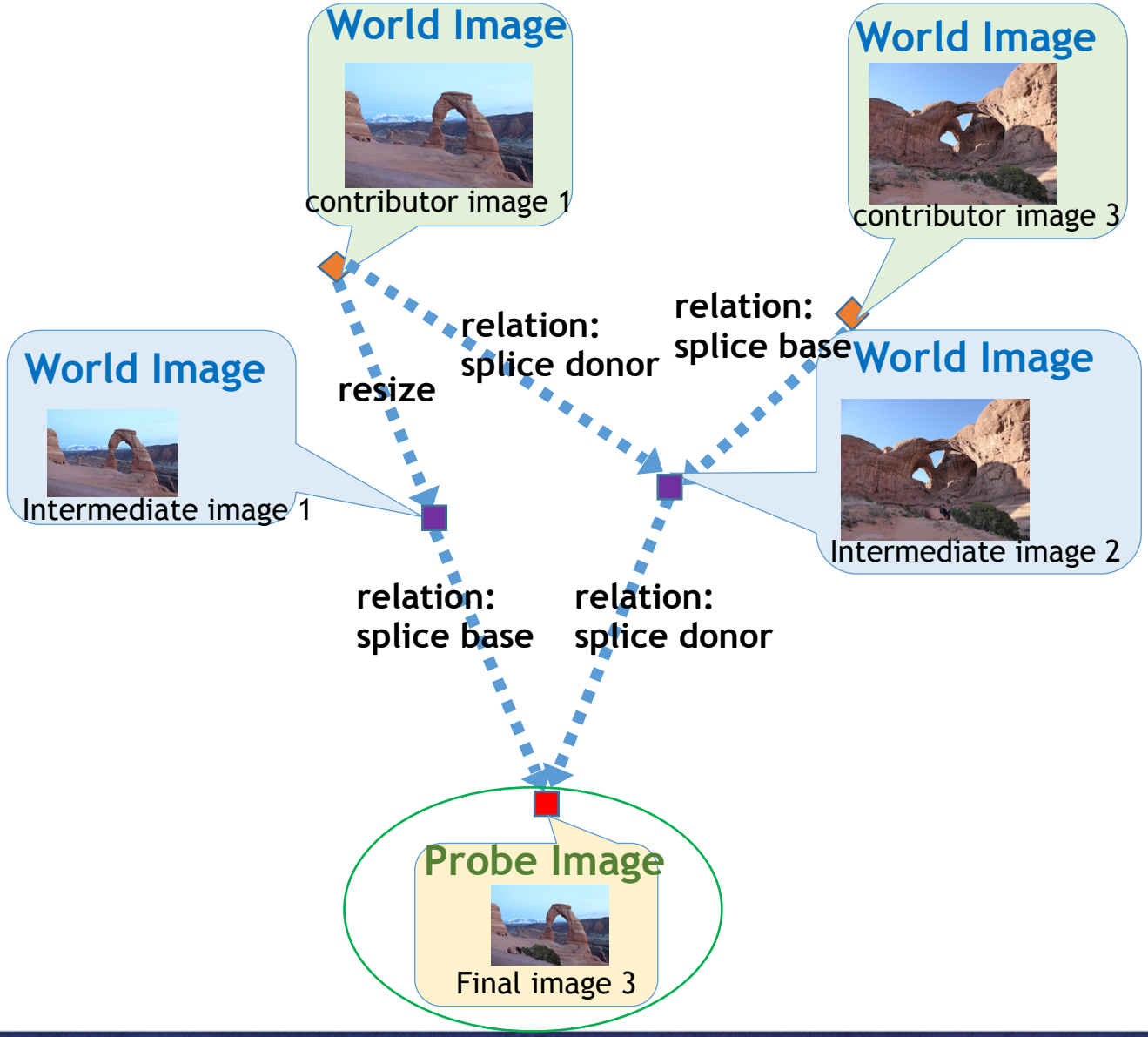
- Sim(nodes)
- Sim (links)
- Sim(nodes+links)

Customized Metrics

(earliest source)

Cost Function
(1) Customized give an application
(2) Graph Edit Distance

Provenance Graph Building System Output



Nimble Evaluation Protocols

- **NC2016 and NC2017 Dev* data sets are free to use for development**
- **NC2017 Evaluation data must not be used for training**
  - Any machine learning or statistical analysis algorithm should complete training, model selection, and tuning prior to performing the task.
- **Trial Independence:** Each trial must be processed independently
  - System output result of a trial does not in any way depend on other trials or other media or media sets in evaluation testing dataset.

NC2017 Evaluation Schedule

Dates	Development Resources
January 27, 2017	•NC2017 Registration opens; Evaluation Plan; Scoring software; Dev2 available
February 27, 2017	•NC2017 Dev2 available
March 29, 2017	•NIST Dry Run scores returned •NC2017 Dev3 released •NIST sends World Data
April 10, 2017	•Encrypted Evaluation World Data distributed
April 12, 2017	•NIST sends World Data decryption Key
April 26, 2017	**•Team Submissions due for: Manipulation Detection and Localization; Splice Detection and Localization; Provenance Filtering; End-to-End Provenance Graph**
April 27, 2017	•Oracle Provenance Graph Data Distributed
May 04, 2017	**•Team submissions due for Oracle Provenance Graph**
May 10, 2017	•Scores released to participants
July 2017	**•DARPA PI Meeting**

- Evaluation website: https://www.nist.gov/itl/iad/mig/nimble-challenge-2017-evaluation

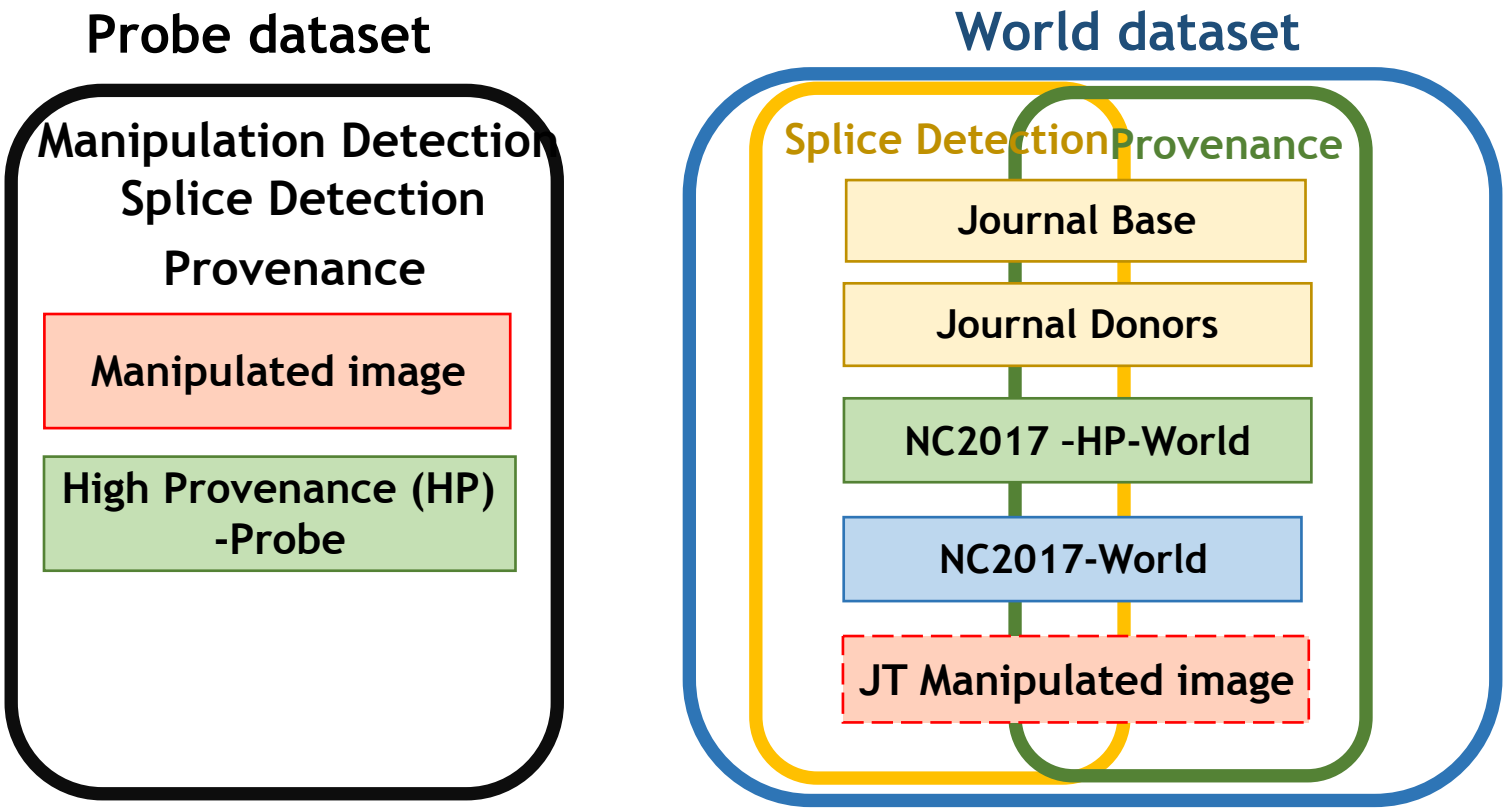
NC2017 Dataset

List of Datasets

	Release Date	Image					Videos	
		Manipulation Journals	MDL Probes	SDL Probes	Proven. Probes	World	Manipul. Journals	Probes
Dev. 1 B.4	02/27/17	50	514	530 K	65	10K	—	—
Dev. 2	**02/27/17**	**199**	**759**	**870 K**	**259**	**100 K**	**20**	**209**
Dev. 3	03/23/17	130 auto 20 prov.	2256	500 K	2156	4092	5	5
NC17_Dev	04/17	394	3563	1 M	2528	≈ 115K	25	214
NC17_Eval	04/12/17	—	10 K	1 M	2991	1 M	—	1083

MDL: Manipulation Detection and Localization
SDL: Splice Detection and Localization
NC17_Dev: the combination of Dev 1, 2, and 3.

NC Dataset Structure Overview
(…/probe, …/world)



NC2017 Dataset Index Files (…/indexes)

- The index file defines the trials for each task.
- Each task has its own index file.
  - Row definition: each row is a trial
- Manipulation detection (Provenance) Input:
  - Column definition: one input image
    - TaskID|ProbeFileID|ProbeFileName|ProbeWidth|ProbeHeight
  - Provenance filtering trials share the same NC2017 evaluation world dataset.
  - Video manipulation detection task follows the same format.
- Splice detection:
  - Column definition: two input images
    - TaskID|ProbeFileID|ProbeFileName|ProbeWidth|ProbeHeight|DonorFileID|DonorFileName|DonorWidth|DonorHeight

NC2017 Dataset Reference
(…/reference)

- The reference files define the ground-truth for evaluation.
- Each task has its own reference subfolder.
- Manipulation/Splice Detection share the same structure
  - mask subfolder: manipulated reference mask
  - NC2017-manipulation-ref.csv
  - NC2017-manipulation-ref-journalmask.csv - support selective scoring
  - NC2017-manipulation-ref-probejournaljoin.csv
- Provenance
  - under development

Evaluations

Metrology Outline

- MediScore Overview
- Manipulation and Splice Task Evaluation
  - Detection and Localization Scoring
  - Selective (Query-based) Scoring
- Proposed Provenance Task Evaluation
  - Provenance Filtering Scoring
  - Provenance Graph Building Scoring

MediScore Overview

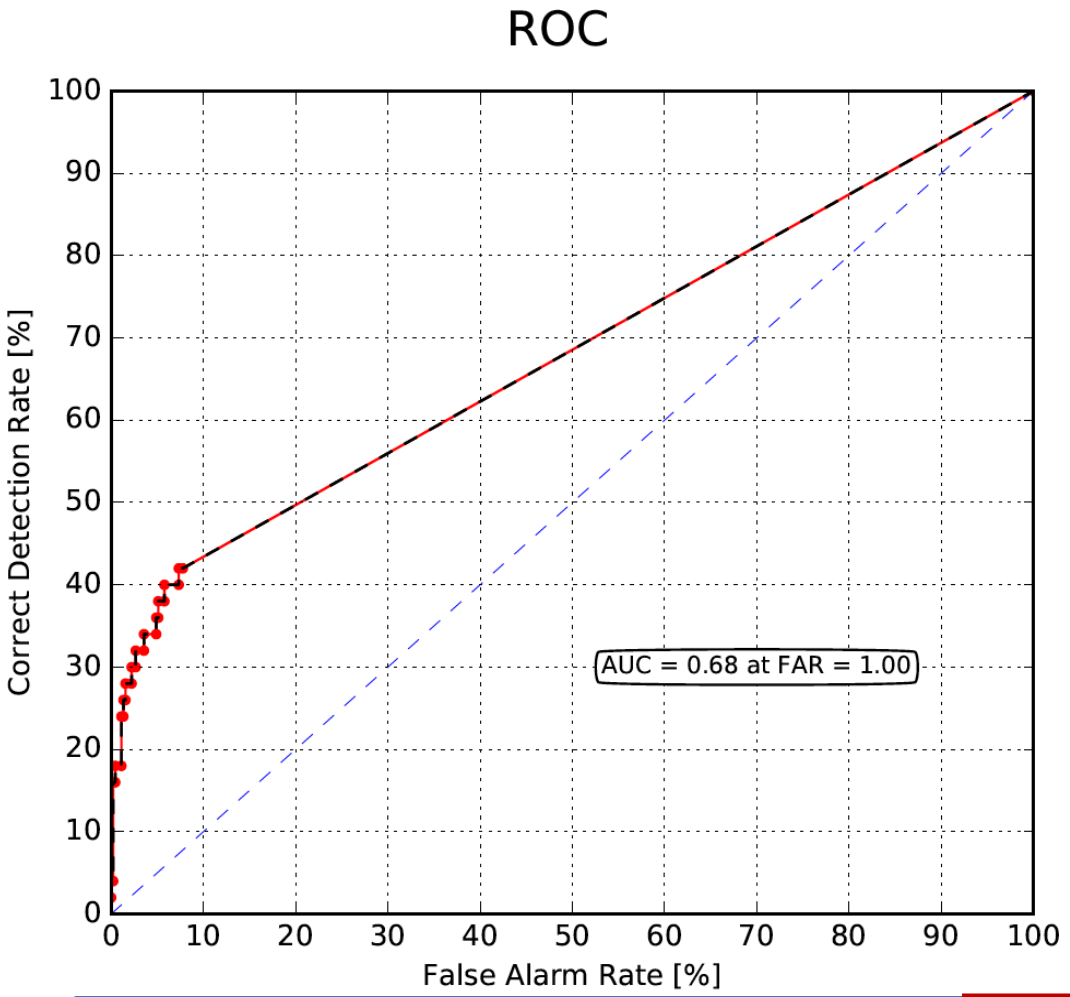
- Written in Python (R version will be no longer supported)
- Tasks supported
  - Manipulation
  - Splice
- Three utilities provided
  - Validation Tool
  - Detection Scorer
  - Mask Scorer (Localization)
- Evaluation Design
  - Self-evaluation
  - Query-based (selective) evaluation

Manipulation and Splice Task Evaluations

Detection Scorer

- Evaluate the accuracy of a system output (e.g., confidence score) to a reference csv file for the multimedia forensic tampering detection
- Evaluation metrics
  - AUC (Area Under Curve) of ROC (receiver operating characteristic)
  - EER (Equal Error Rate) with DET (detection error tradeoff)

$ python DetectionScorer.py -t manipulation -r inRef -x inIndex -s inSys [OPTIONS]



Baseline: Copymove

Dataset: NC2017

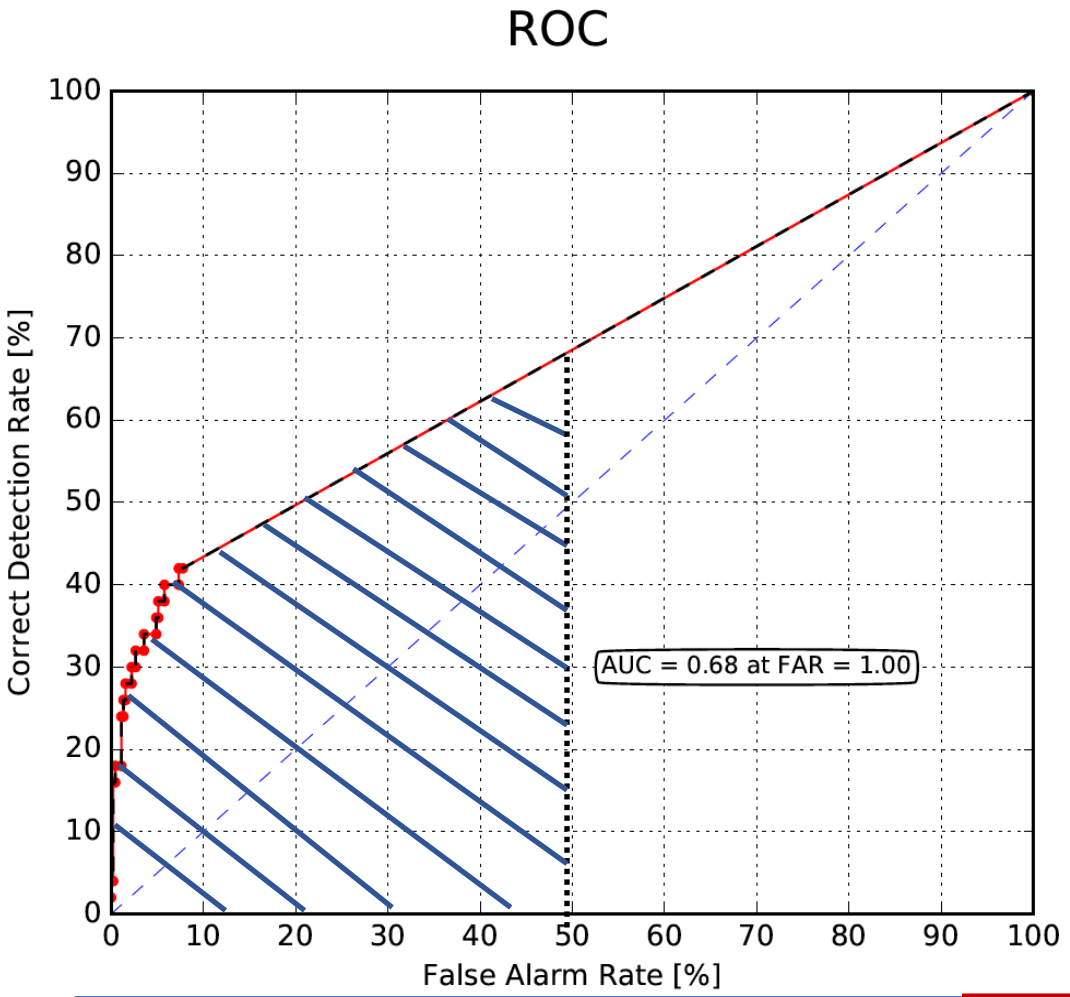
- Total trials: 500
- Manipulations: 50

Bootstrapping (500 times)

- 90% confidence interval
- Lower bound: 0.05
- Upper bound: 0.95

AUC	FAR_STOP	EER	AUC_CI_LOWER	AUC_CI_UPPER
0.679533	1	0.328889	0.620826	0.735491

$ python DetectionScorer.py -t manipulation -r inRef -x inIndex -s inSys [OPTIONS]



Baseline: Copymove

Dataset: NC2017

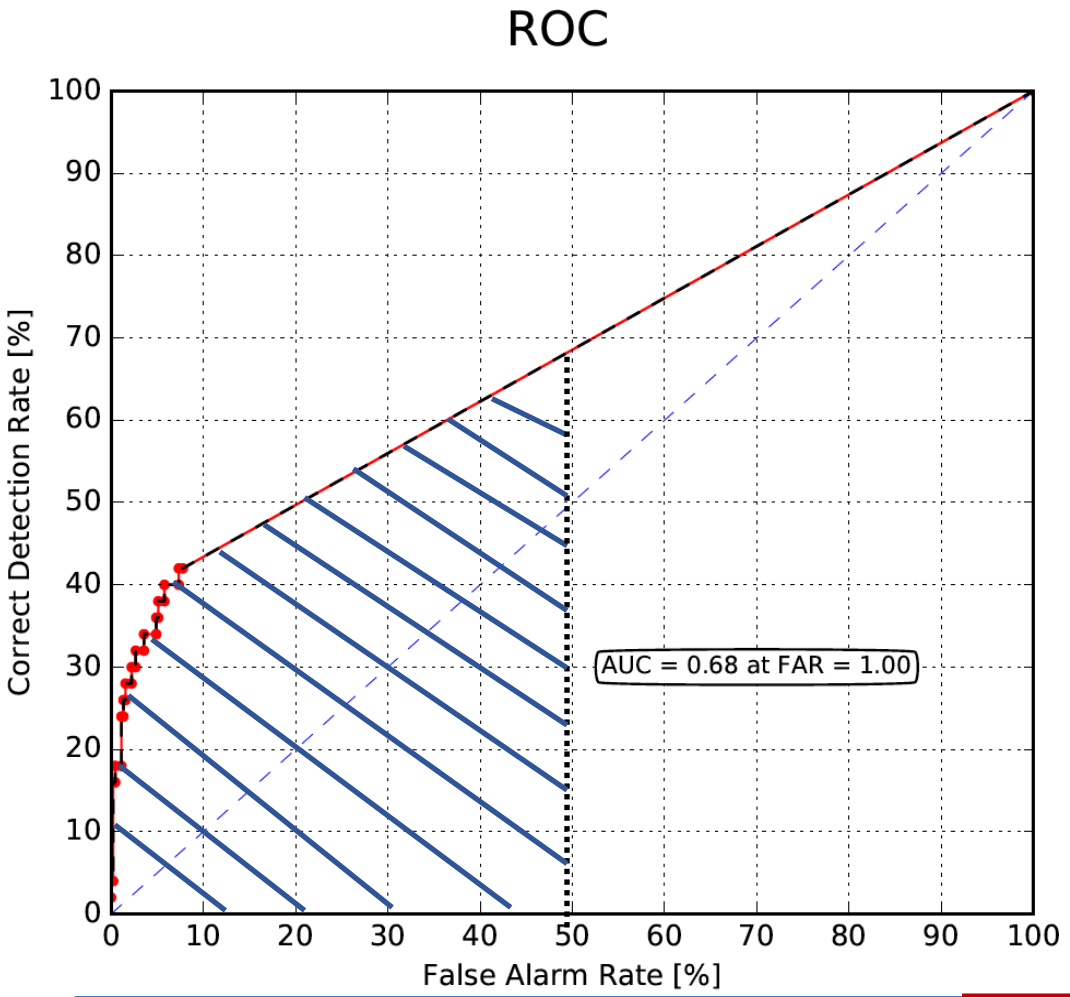
- Total trials: 500
- Manipulations: 50

Bootstrapping (500 times)

- 90% confidence interval
- Lower bound: 0.05
- Upper bound: 0.95

AUC	FAR_STOP	EER	AUC_CI_LOWER	AUC_CI_UPPER
0.679533	1	0.328889	0.620826	0.735491

$ python DetectionScorer.py -t manipulation -r inRef -x inIndex -s inSys [OPTIONS]



Baseline: Copymove

Dataset: NC2017

- Total trials: 500
- Manipulations: 50

Same metrics for Splice task

Bootstrapping (500 times)

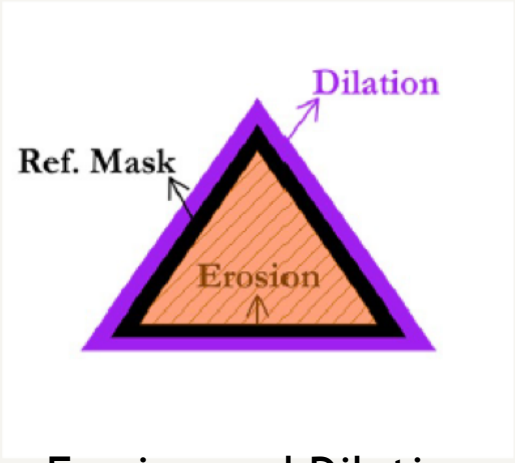
- 90% confidence interval
- Lower bound: 0.05
- Upper bound: 0.95

AUC	FAR_STOP	EER	AUC_CI_LOWER	AUC_CI_UPPER
0.679533	1	0.328889	0.620826	0.735491

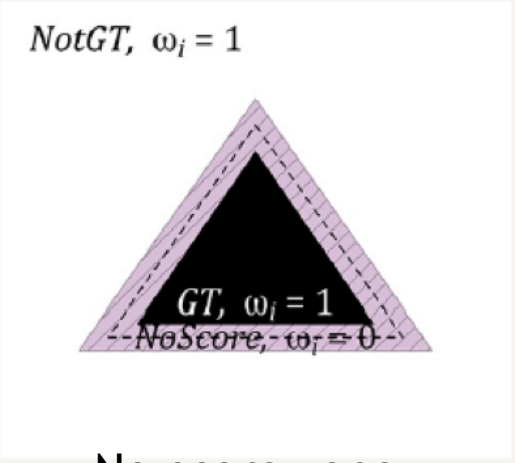
Localization (Mask) Scorer

- Evaluate the accuracy of a system output mask to a reference mask (localization)
  - Evaluate on the target trials only (manipulations)
    - If the system output mask for a trial was not provided, the worst score (e.g.,-1) will be given for that trial
  - Evaluation metrics (for both binary and grayscale)
    - ***MCC (Matthews Correlation Coefficient)***
    - NMM (Nimble Mask Metric)
    - WL1 (Weighted L1 Loss)
      - Binary: BWL1
      - Grayscale: GWL1

Reference Mask

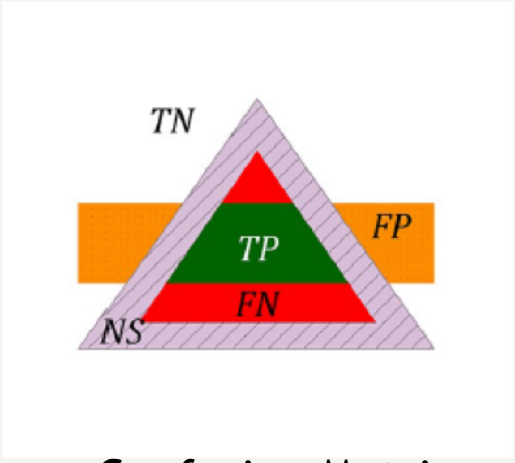


Erosion and Dilation for weights



No-score zone (weights)

System Output Mask



Confusion Matrix

Localization Evaluation Metrics

- MCC [-1, 1]
  - $\text{MCC} = \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$
- NMM [-1, 1]
  - $\text{NMM(Gray)} = \max\left\{\frac{\sum_{i\in GT}(2*M_S(i)-1)-\sum_{i\in NotGT} M_S(i)}{\text{size}(GT)}, -1\right\}$
  - $\text{NMM(Binary)} = \max\left\{\frac{\text{size}(TP)-\text{size}(FN)-\text{size}(FP)}{\text{size}(GT)}, -1\right\}$
- WL1 [0, 1]
  - $\text{WL1}\left(\widehat{M_r}, \widehat{M_s}\right) = \frac{1}{\text{size}(GT)}\sum_{i=1}^{N}\omega_i \frac{|\widehat{M_r}(i)-\widehat{M_s}(i)|}{255}$
- Grayscale system output mask
  - 1) the algorithm provides a threshold to binalize the output mask
  - 2) the scoring tool computes a given metric through all possible thresholds (in the output mask) and report the best score

```
$ python MaskScorer.py –t manipulation -r inRef -x inIndex -s inSys [OPTIONS]
```

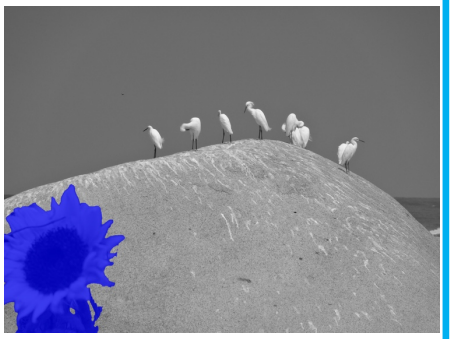
- Summary report (CSV files)
  - Metric scores per trial
  - Average over trials
  - Metadata summaries
- Manipulation: visualization per mask (-html)
  - Binary mask example
  - Grayscale mask example

```
$ python MaskScorer.py -t splice -r inRef -x inIndex -s inSys [OPTIONS]
```

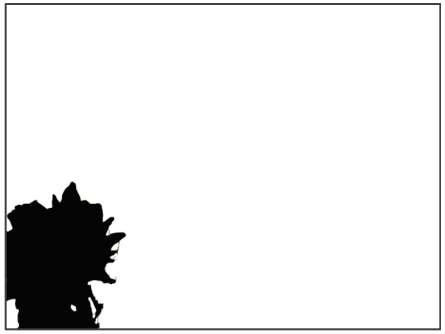
Splice: visualization per mask (-html)

Probe Mask Evaluation

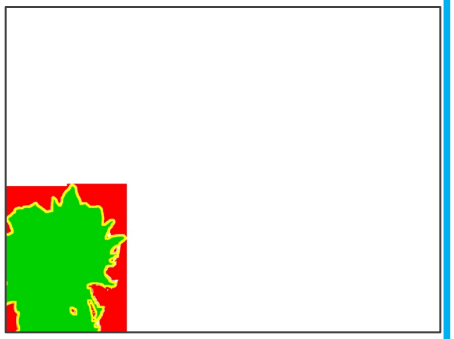
Manipulated image



Composite color mask



System output mask

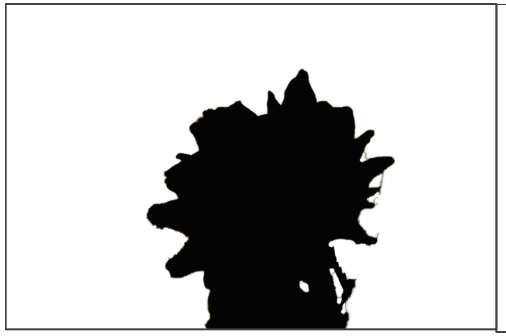


Result visualization

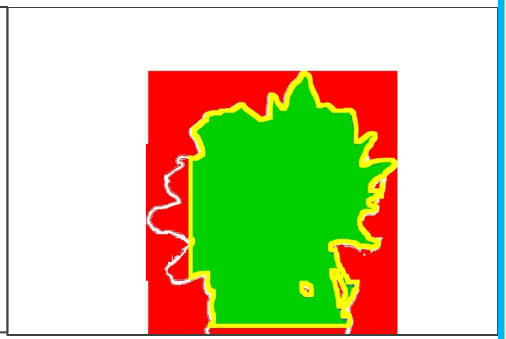
Donor Mask Evaluation

Donor image

Composite color mask



System output mask



Result visualization

Selective (Query-based) Evaluation

- Issue
  - Allow researchers **to solve their individual problems**
    - A system should not be penalized, even if the system focuses on detecting specific manipulation evidence
      - e.g., double JPEG detector does not work on PNGs
    - Need for estimating performance on specified operations only
  - NIST needs to know **what to evaluate for that system.**
  - We must **balance** – reasonable accommodations for the system with gaming the evaluation protocol
    - e.g., a per-trial designation at evaluation time could be abused

Selective (Query-based) Evaluation

- Approach
  - **Performers declare the limits/presumptions** of their system before scoring
    - within a system description or a special file
  - NIST populates the reference data with information and **enables selective evaluation** (via proper trial selection)
    - **Selected trials** (based on the metadata queried) will be scored, while **unselected trials** will NOT be scored.

Query-based Evaluation (Detection Scorer)

- Evaluate algorithm performance on either subsets or partitions of the data set based on the user-specified queries
  - Query (one or multiple queries)
    - Filters both target and non-target trials and processes scoring run per query
  - Query for partitions (single query)
    - Separates (automatically) the data set into M partitions by filtering both target and non-target trials and processes one or multiple scoring runs
  - Query for selective manipulations (one or multiple queries)
    - Restricts filtering to target trials only (while using all non-target trials) and processes scoring run per query

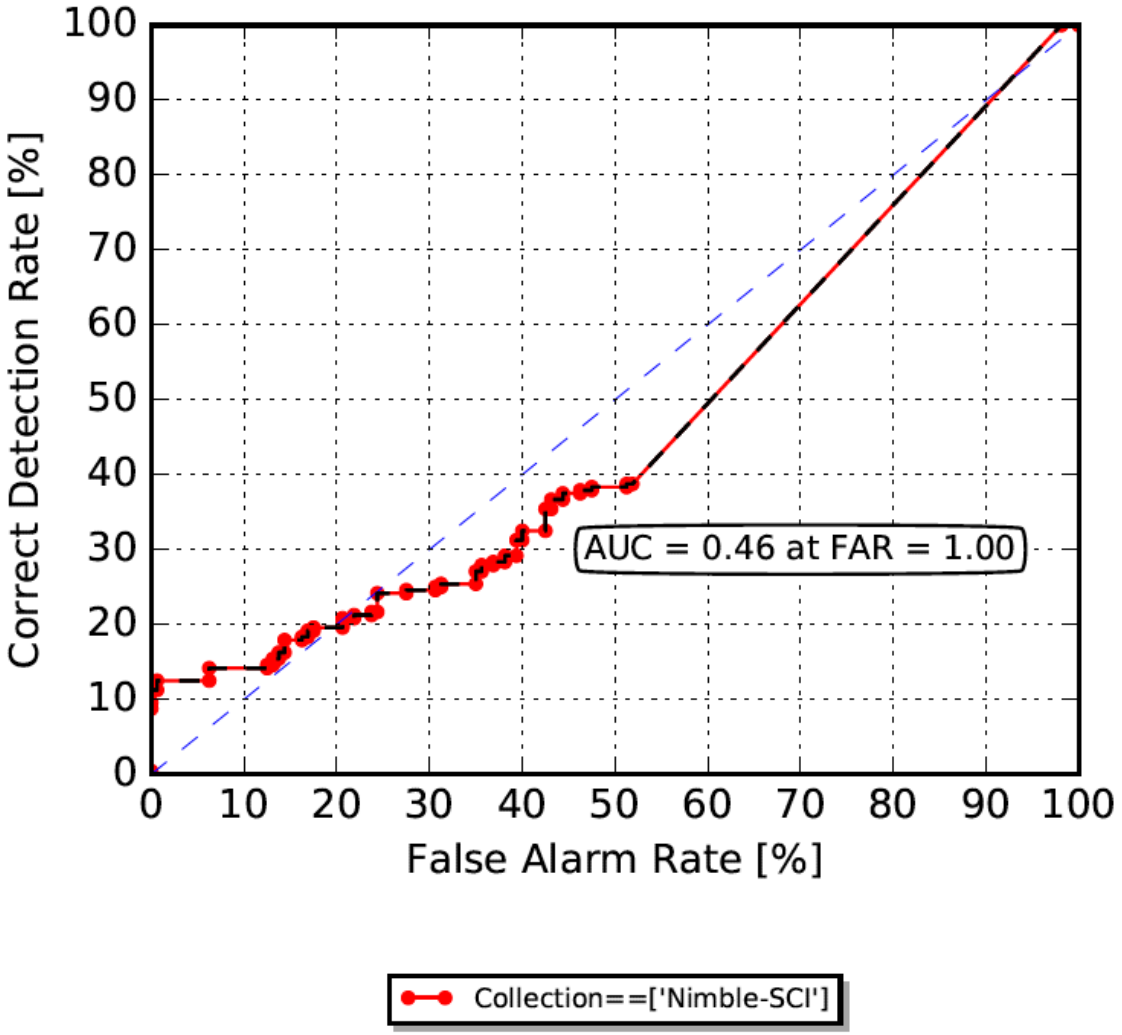
Query
(-q, --query)

-q "Collection==['Nimble-SCI']"

Baseline: DCT02

Dataset: NC2016

Question: What is the algorithm performance on the NIMBLE-SCI dataset only?

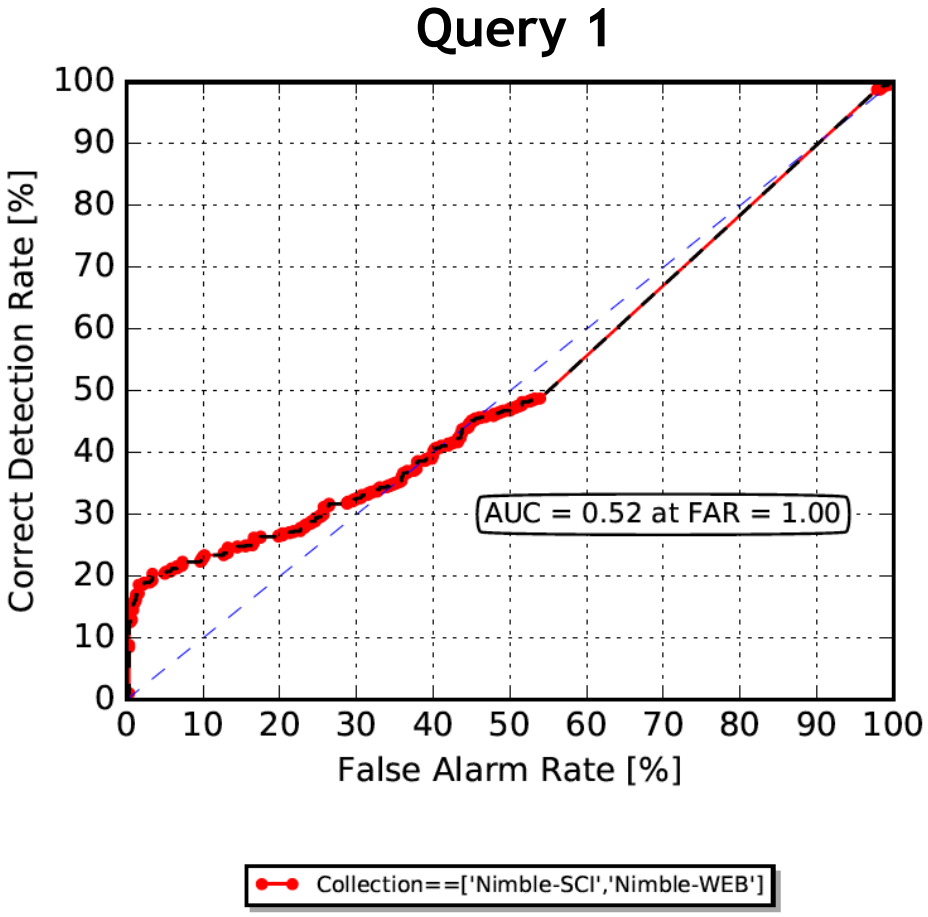


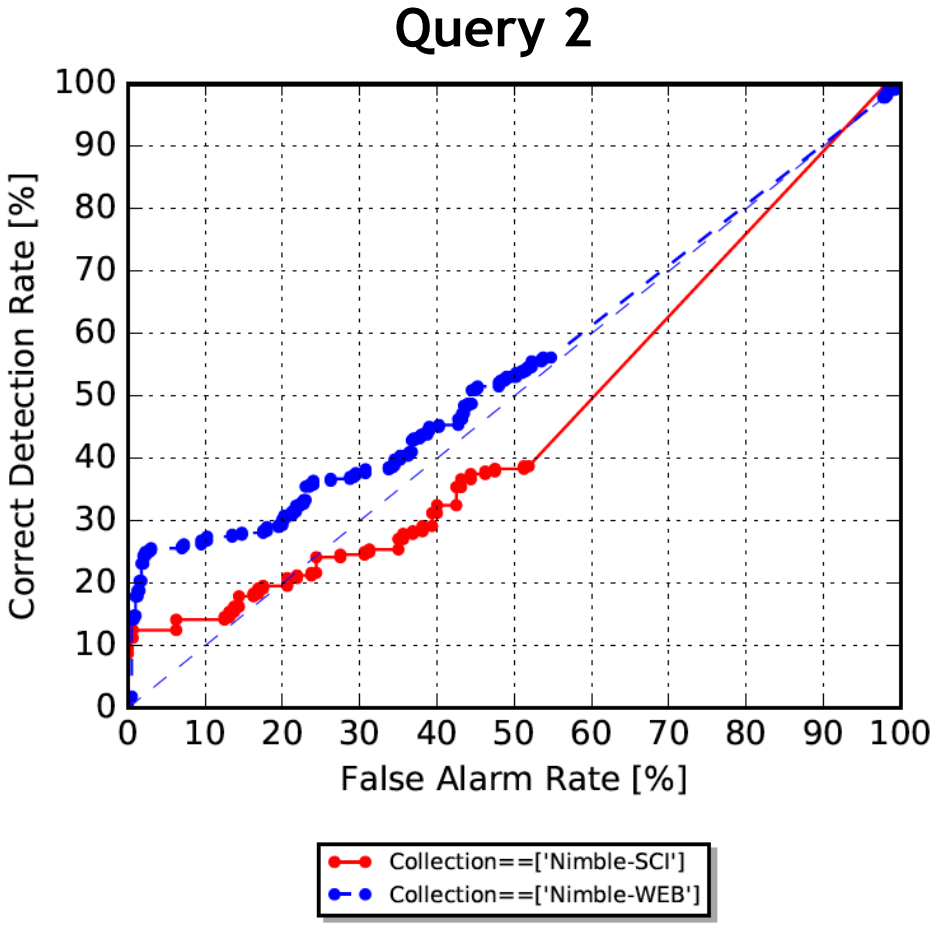
Query	AUC	FAR_STOP	EER	AUC_CI_LOWER	AUC_CI_UPPER
Collection==['Nimble-SCI']	0.462995	1	0.565625	0.411921	0.502428

Query 1	-q "Collection==['Nimble-SCI','Nimble-WEB']"
Query 2	-q "Collection==['Nimble-SCI']" "Collection==['Nimble-WEB']"

Query 1 -q "Collection==['Nimble-SCI','Nimble-WEB']"

Query 2 -q "Collection==['Nimble-SCI']" "Collection==['Nimble-WEB']"





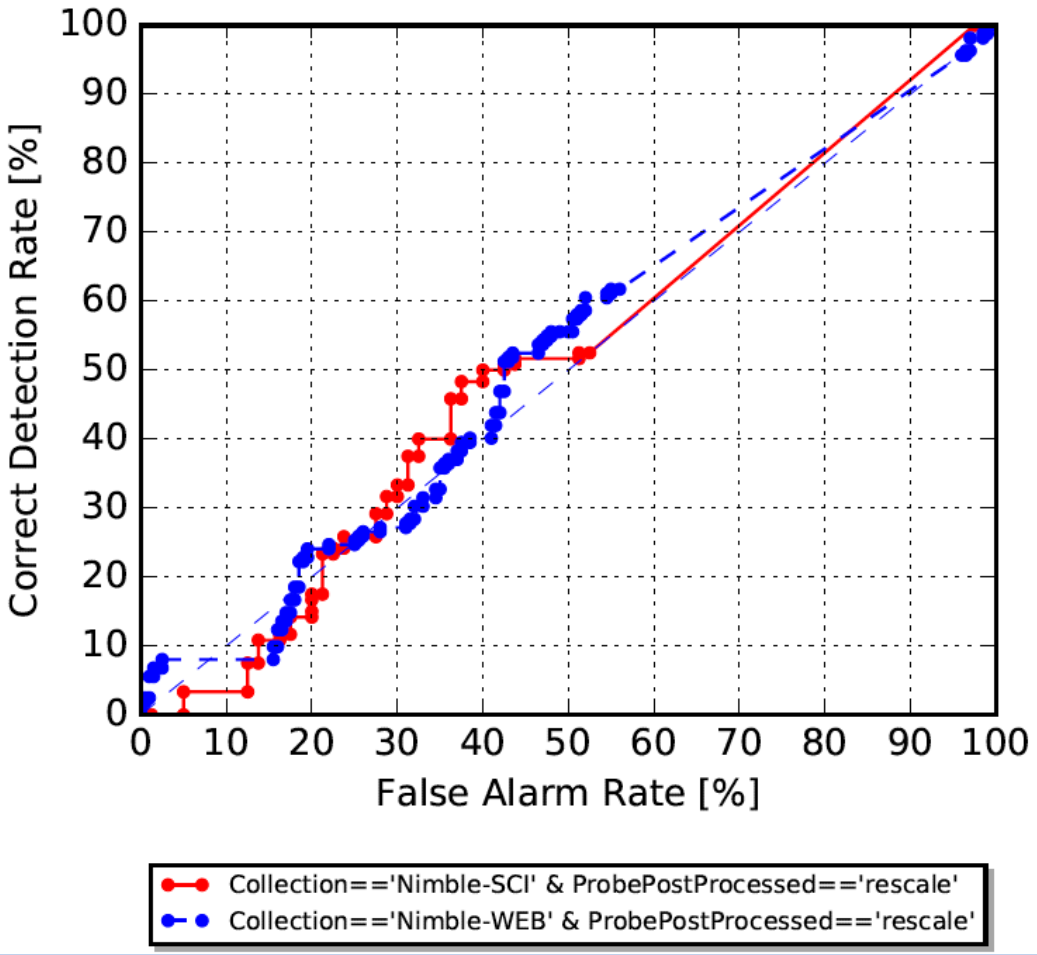
Question: What is the performance on one dataset that contains both Nimble-SCI and Nimble-WEB?

Question: What is the comparison of performance between the two separated datasets?

Query for Partitions (-qp, --queryPartition)

-qp "Collection==['Nimble-SCI','Nimble-WEB'] &
ProbePostProcessed==['rescale']"

Question:
How do the different datasets (e.g., Nimble-SCI and Nimble-WEB) behave **after** applying the post processing technique (rescale)?

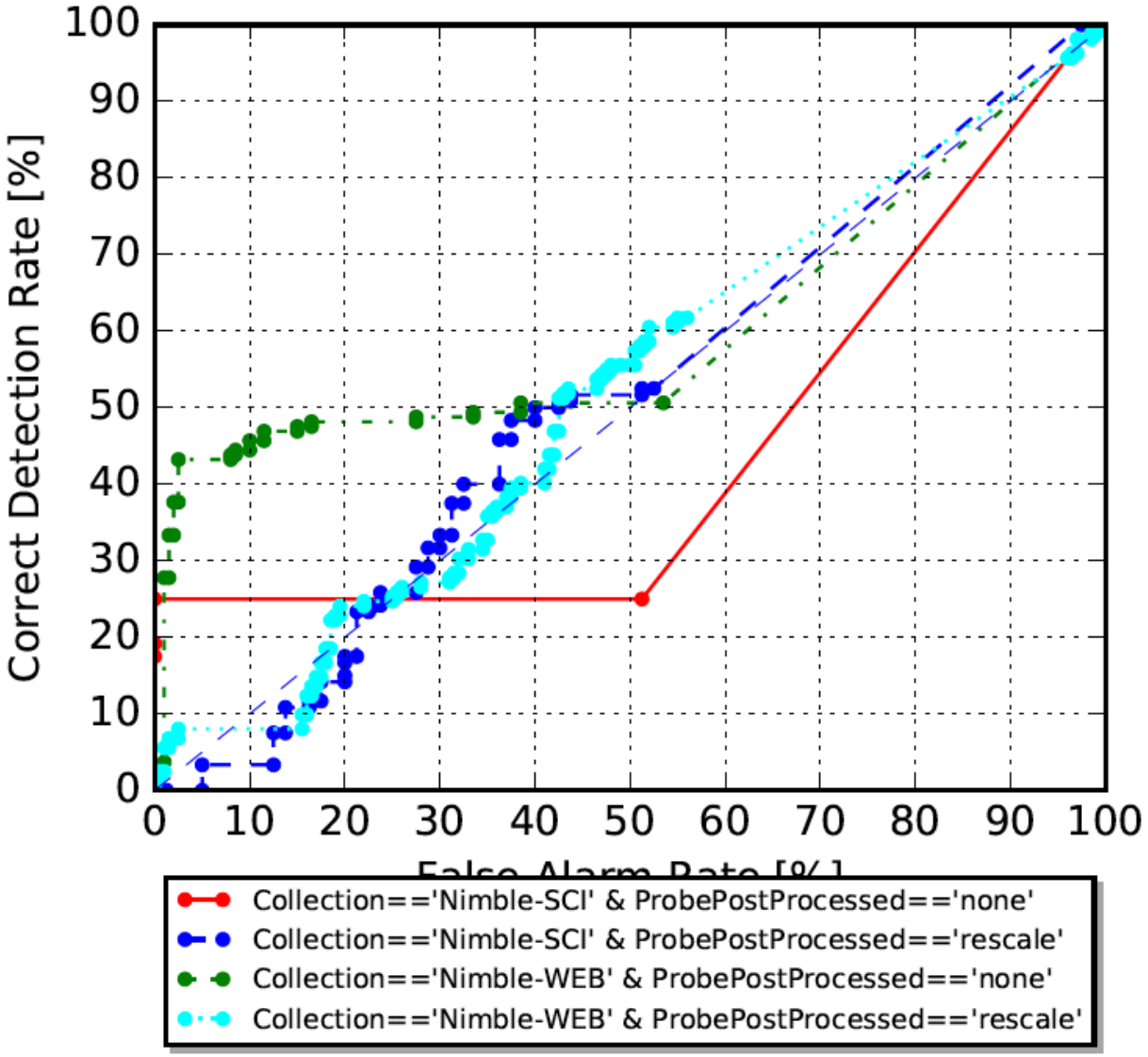


	Collection	ProbePostProcessed	auc	fpr_stop	eer	auc_ci_lower	auc_ci_upper
Partition_0	'Nimble-SCI'	'rescale'	0.510521	1	0.497917	0.441763	0.574645
Partition_1	'Nimble-WEB'	'rescale'	0.519691	1	0.463981	0.471256	0.573472

```
-qp "Collection==['Nimble-SCI','Nimble-WEB'] &
ProbePostProcessed==['none', 'rescale']"
```

-qp "Collection==['Nimble-SCI','Nimble-WEB'] & ProbePostProcessed==['none', 'rescale']"

Question:
How do the different datasets (e.g., Nimble-SCI and Nimble-WEB) behave **before and after** applying the post processing technique (rescale)?



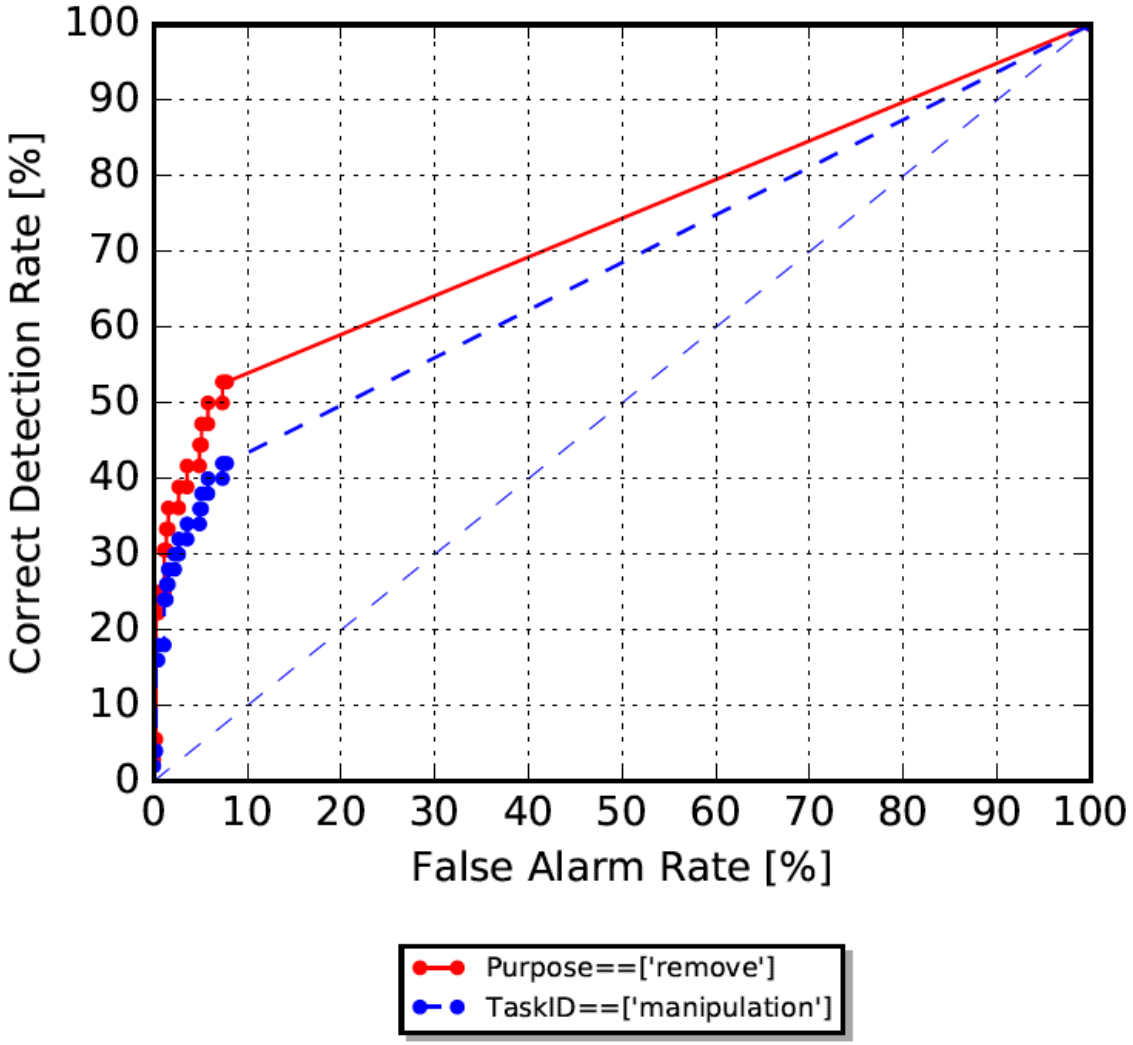
Query for Selective Manipulations (-qm, --queryManipulation)

-qm "Purpose==['remove']" "TaskID==['manipulation'] "

Baseline: Copymove

Dataset: NC2017

Question: What is the comparison of all manipulations versus the only manipulations intended to "removal" imagery?



Same as -q "Purpose ==['remove'] and IsTarget == ['Y'] or IsTarget == ['N']" "TaskID==['manipulation'] and IsTarget == ['Y'] or IsTarget == ['N']"

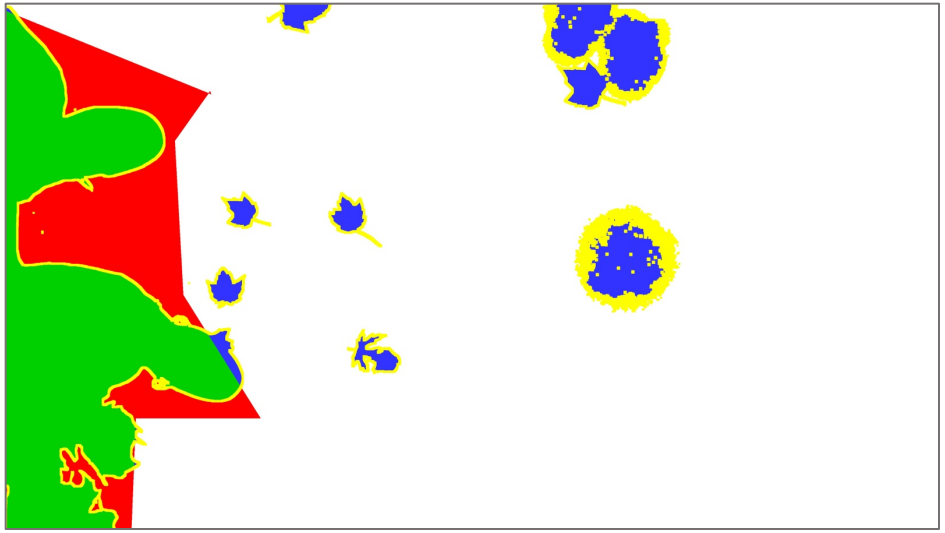
Query-based Evaluation (Mask Scorer)

- Support for all three query options
  - Query (-q –query)
  - Query for partitions (-qp –queryPartitions)
  - Query for selective manipulations (-qm --queryManipulation)
    - Allows only **one query**
    - The selected/unselected metadata are applicable **within a mask**
    - The **unselected operations of the manipulated area** will be the part of the no-score zone

Default 'all'

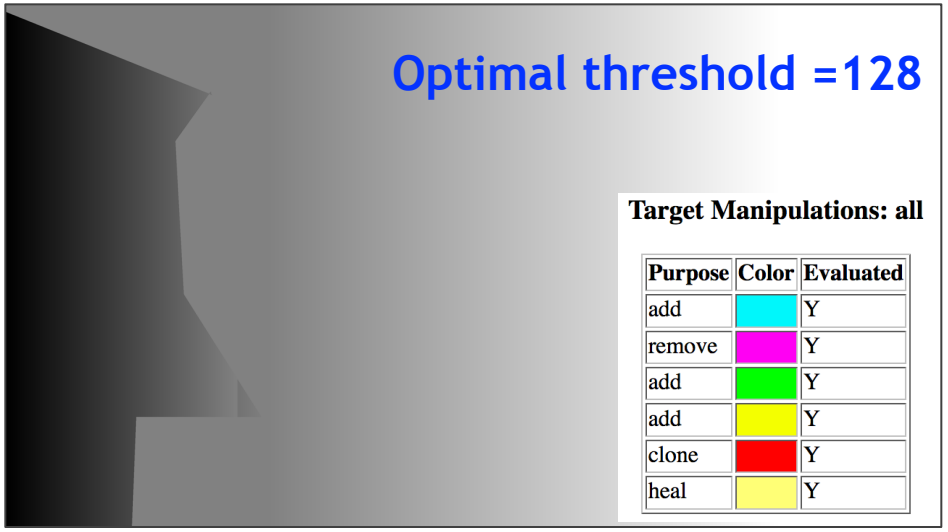
Composite with Color Mask

Evaluation Result Visualization



Manipulation 'all'

System Output Mask



Target Manipulations: all

Purpose	Color	Evaluated
add		Y
remove		Y
add		Y
add		Y
clone		Y
heal		Y

Evaluation Scoring Results

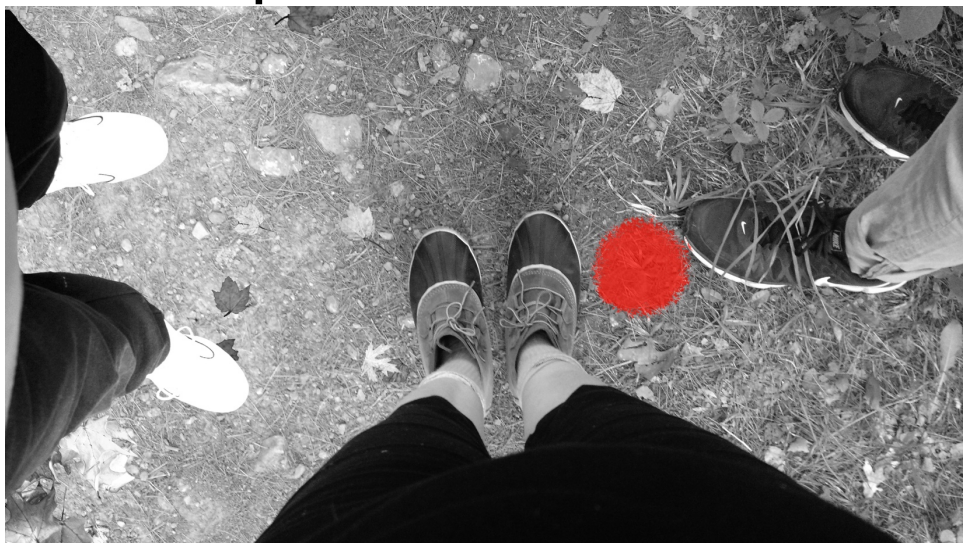
Confusion Measures	Pixels	Proportion
True Positives (TP: green):	1012095	0.107
False Positives (FP: red):	588762	0.062
True Negatives (TN: white):	7542705	0.8
False Negatives (FN: blue):	282147	0.03
Boundary No-Score Zone (BNS: yellow):	308691	0.033
Selective No-Score Zone (SNS: pink):	0	0.0

NIMBLE Mask Metric (NMM): 0.109
Matthews Correlation Coefficient (MCC): 0.65
Binary Weighted L1 Loss (WL1): 0.092
Grayscale Weighted L1 Loss (WL1): 0.267

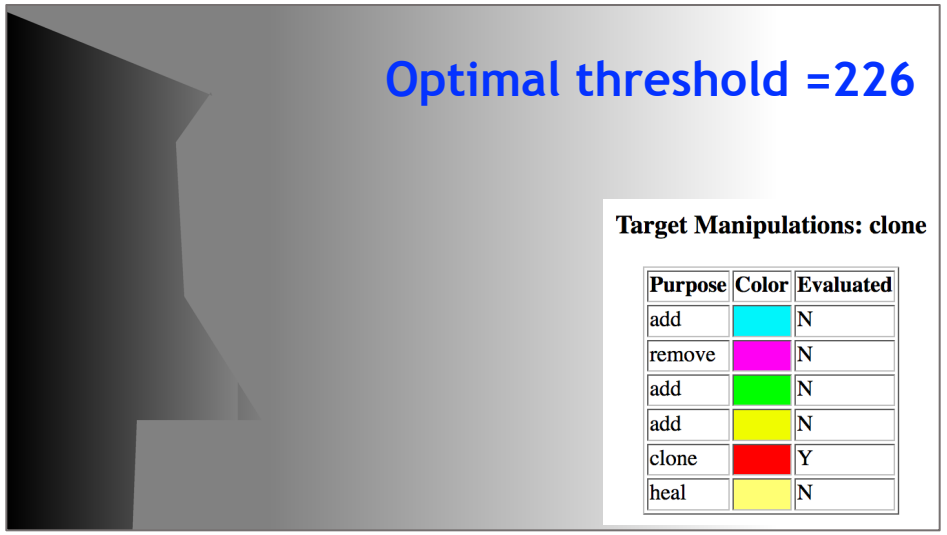
-qm “Purpose==[‘clone’]

Manipulation ‘clone’ only

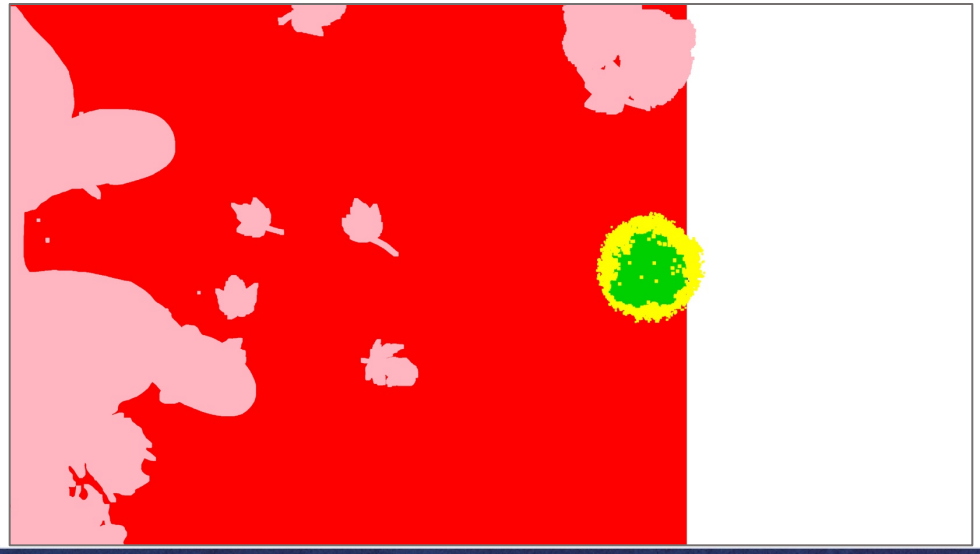
Composite with Color Mask



System Output Mask



Evaluation Result Visualization



Evaluation Scoring Results

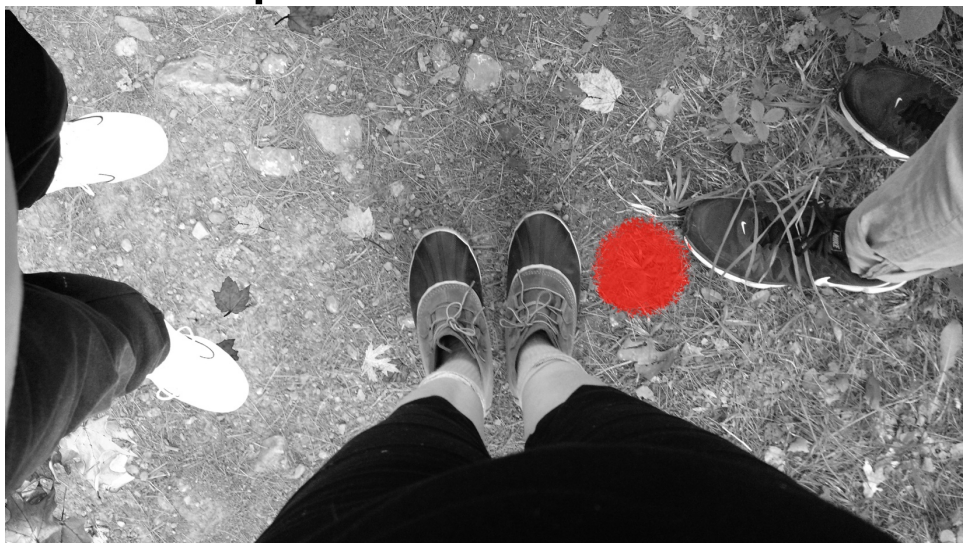
Confusion Measures	Pixels	Proportion
True Positives (TP: green):	72787	0.009
False Positives (FP: red):	5214273	0.639
True Negatives (TN: white):	2868845	0.352
False Negatives (FN: blue):	65	0.0
Boundary No-Score Zone (BNS: yellow):	88199	0.011
Selective No-Score Zone (SNS: pink):	1490231	0.183

NIMBLE Mask Metric (NMM): -1.0
Matthews Correlation Coefficient (MCC): 0.07
Binary Weighted L1 Loss (WL1): 0.639
Grayscale Weighted L1 Loss (WL1): 0.268

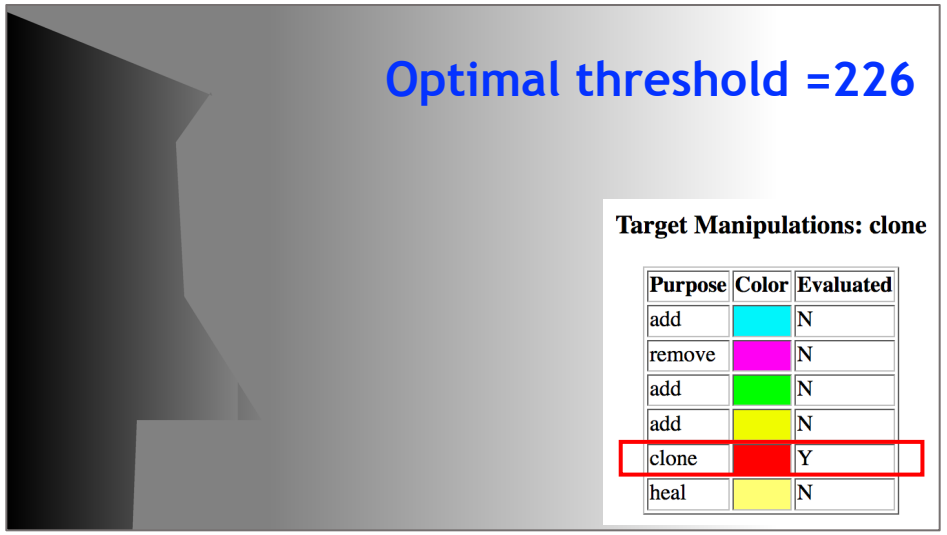
-qm "Purpose==['clone']

Manipulation 'clone' only

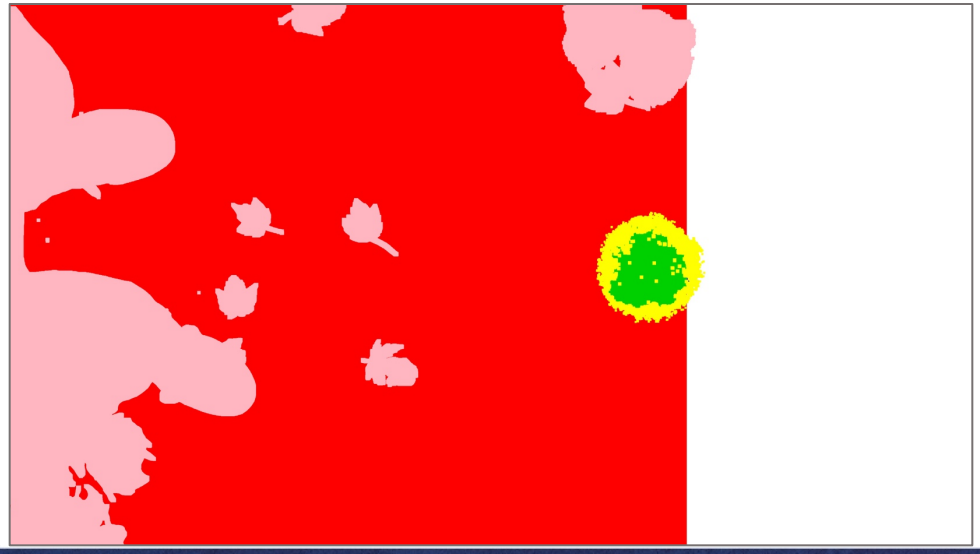
Composite with Color Mask



System Output Mask



Evaluation Result Visualization



Evaluation Scoring Results

Confusion Measures	Pixels	Proportion
True Positives (TP: green):	72787	0.009
False Positives (FP: red):	5214273	0.639
True Negatives (TN: white):	2868845	0.352
False Negatives (FN: blue):	65	0.0
Boundary No-Score Zone (BNS: yellow):	88199	0.011
Selective No-Score Zone (SNS: pink):	1490231	0.183

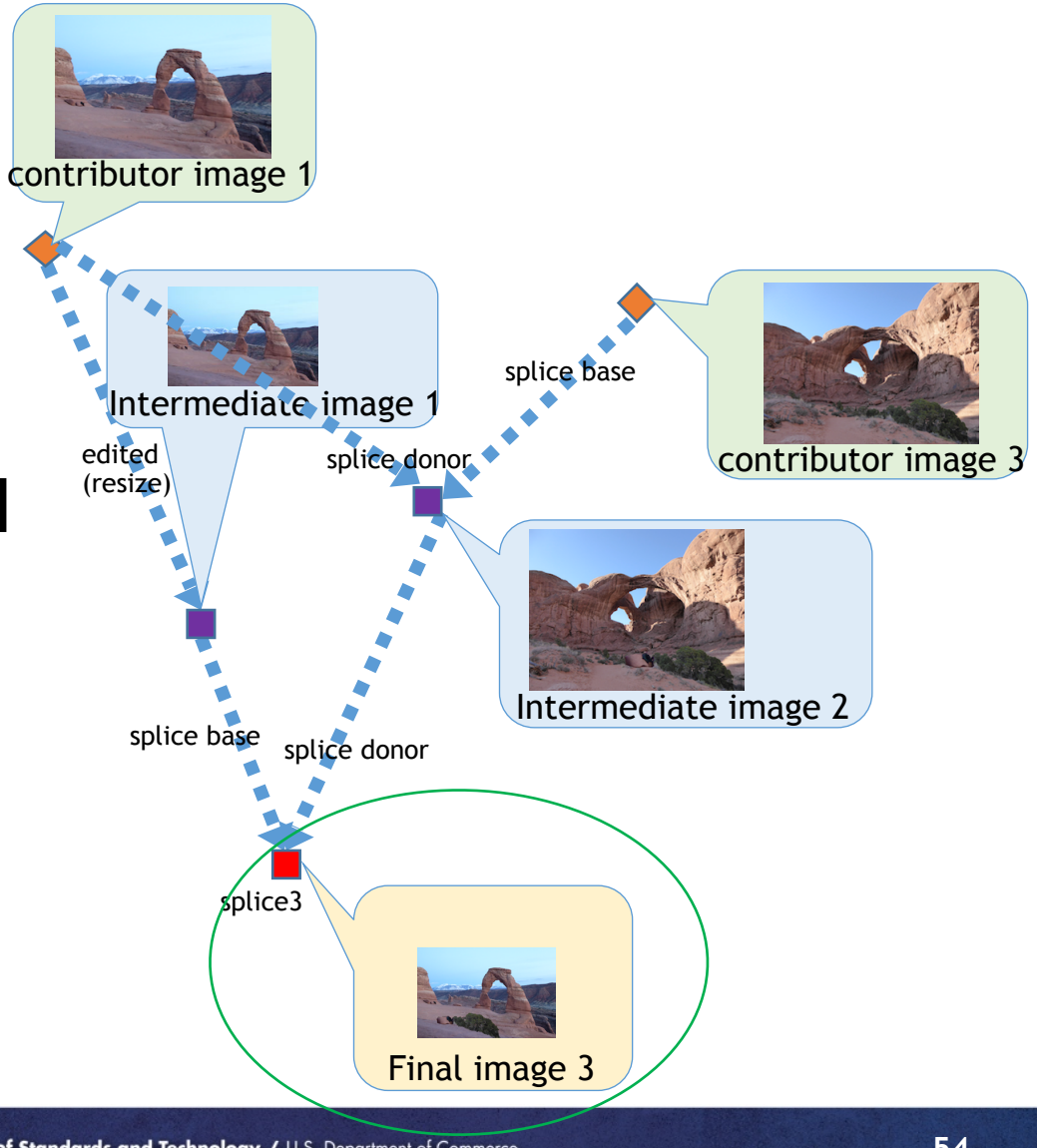
NIMBLE Mask Metric (NMM): -1.0
Matthews Correlation Coefficient (MCC): 0.07
Binary Weighted L1 Loss (WL1): 0.639
Grayscale Weighted L1 Loss (WL1): 0.268

Proposed Provenance Task Metrology

- Provenance ultimate goal and evaluation strategy
- Evaluation protocol: overview
- Evaluation reference graph
- Evaluation metrics

The Ultimate Goal of Provenance Tasks

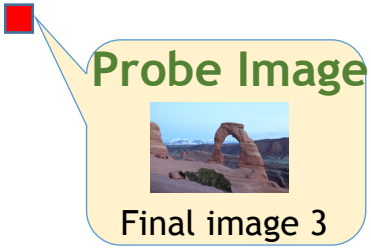
- The ultimate goal is to be able to build the provenance graph
- Task design and metrics design should be consistent with the ultimate goal.



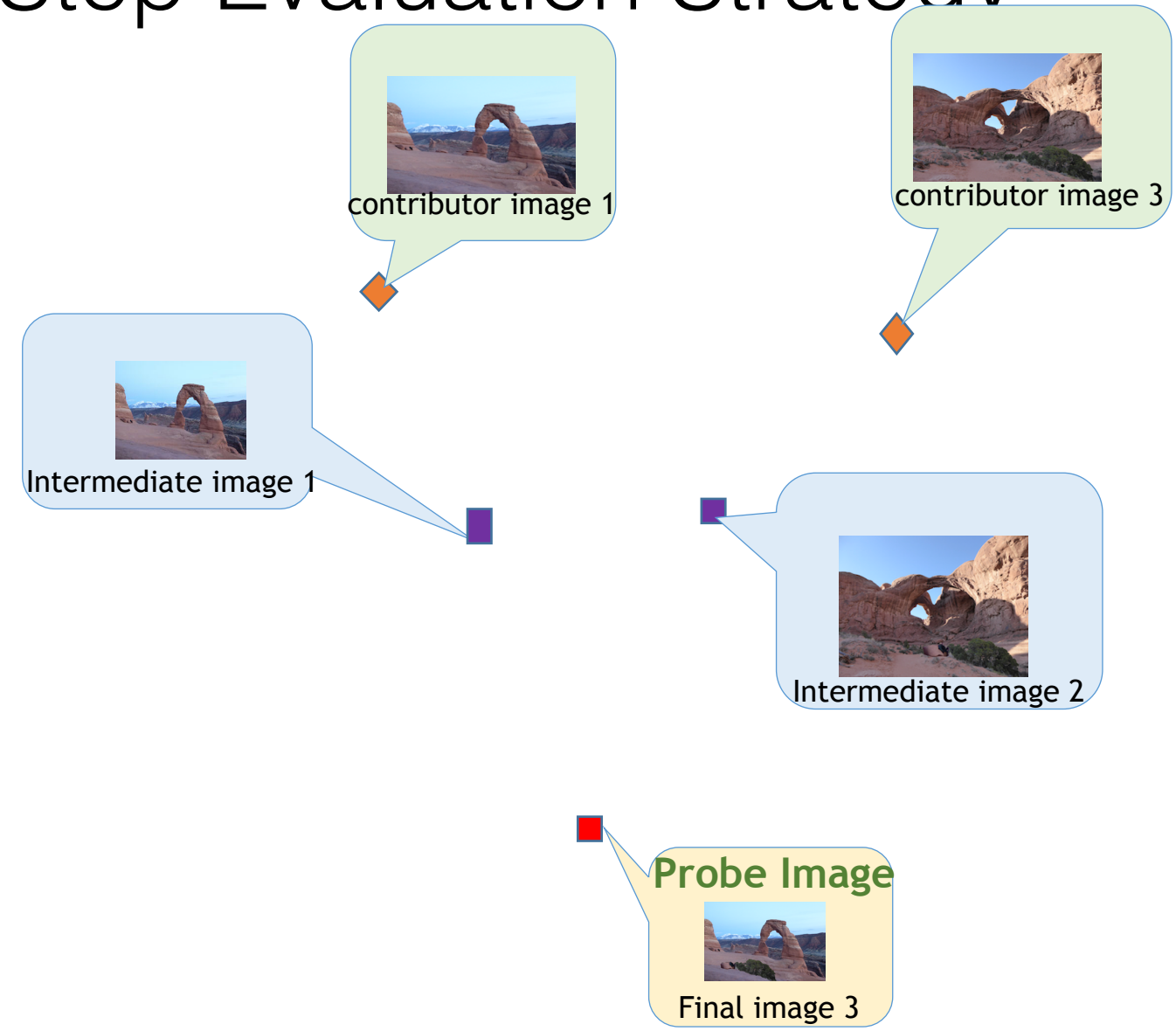
Step by Step Evaluation Strategy

Step by Step Evaluation Strategy

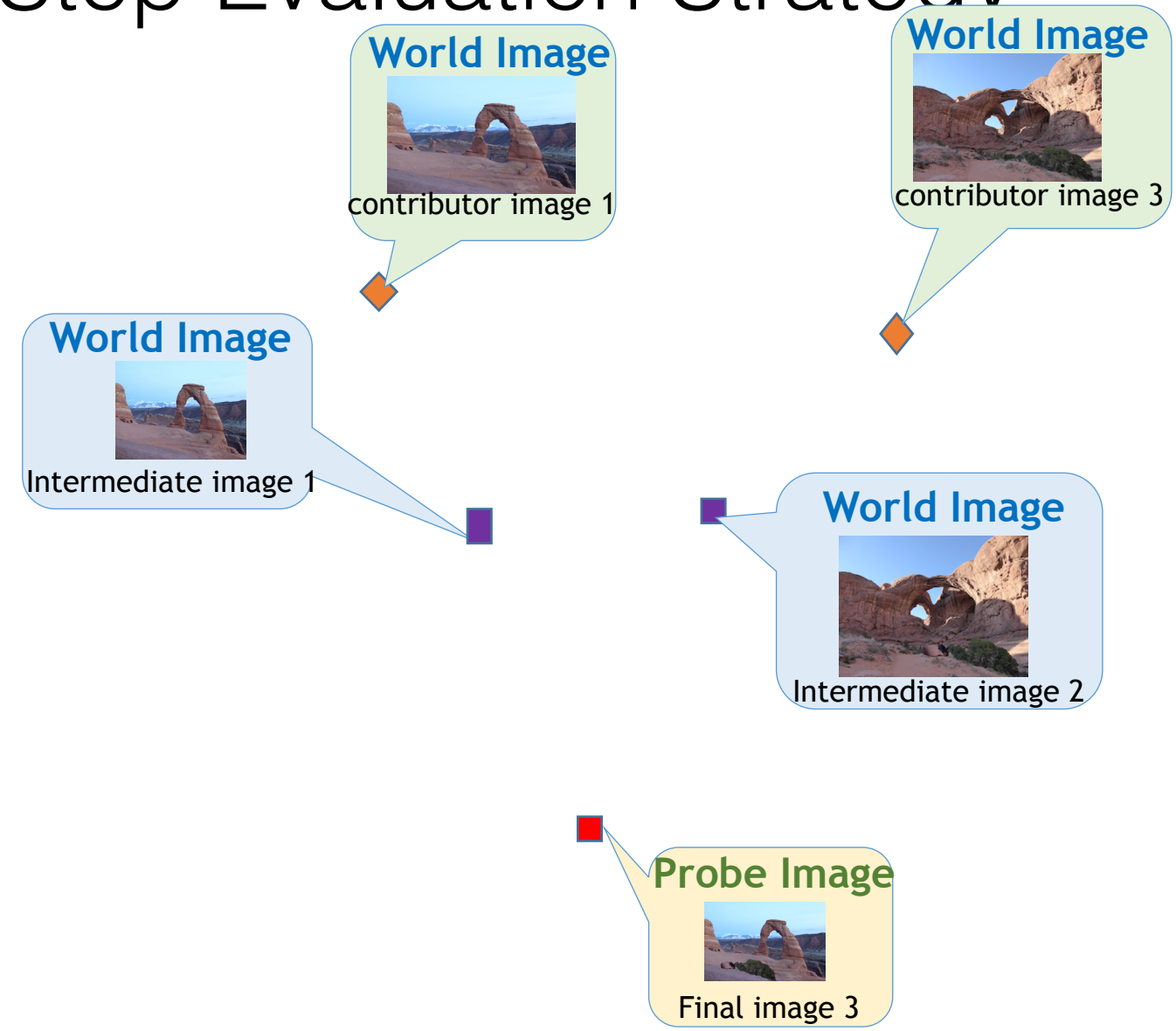
Step by Step Evaluation Strategy



Step by Step Evaluation Strategy

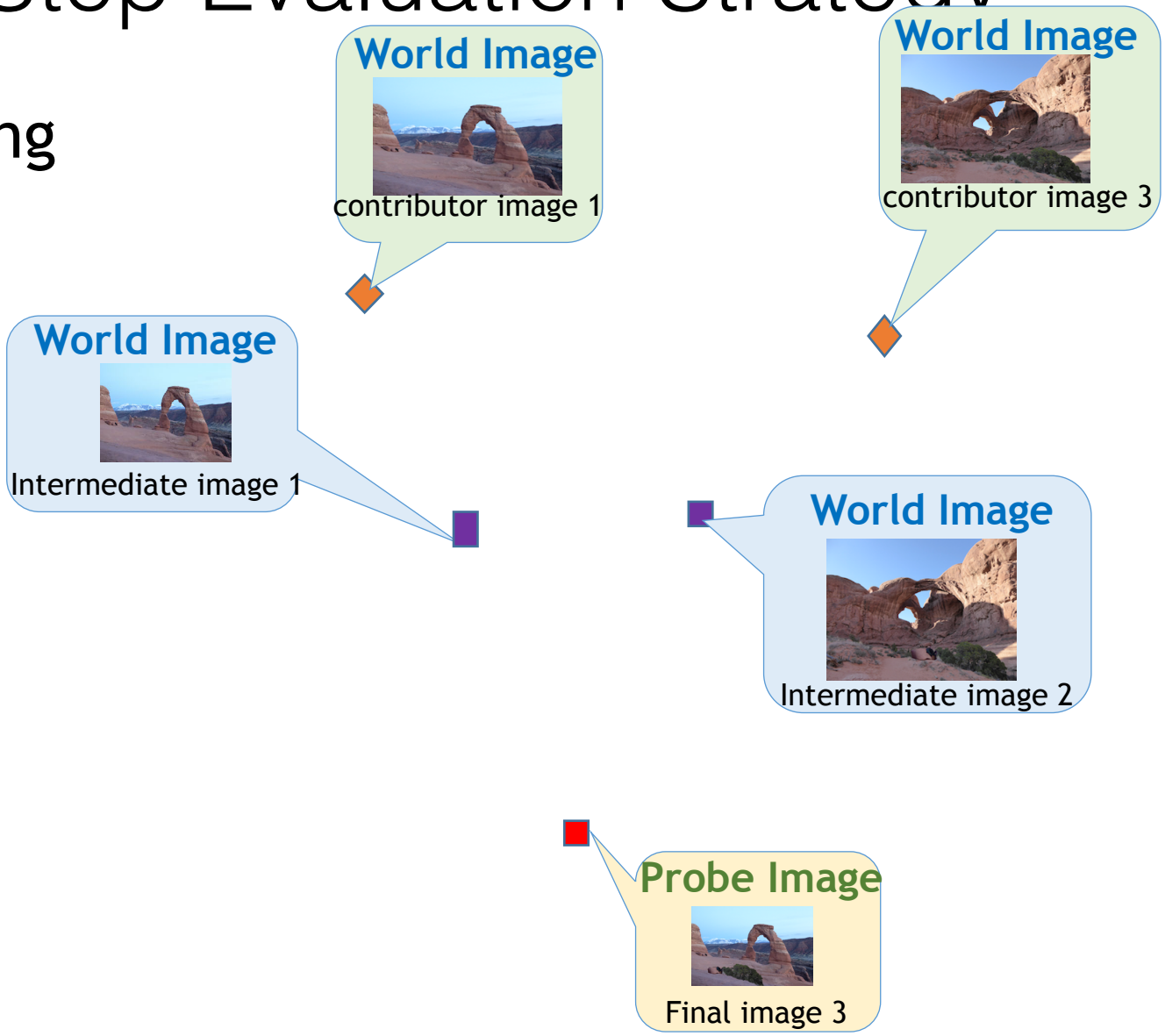


Step by Step Evaluation Strategy



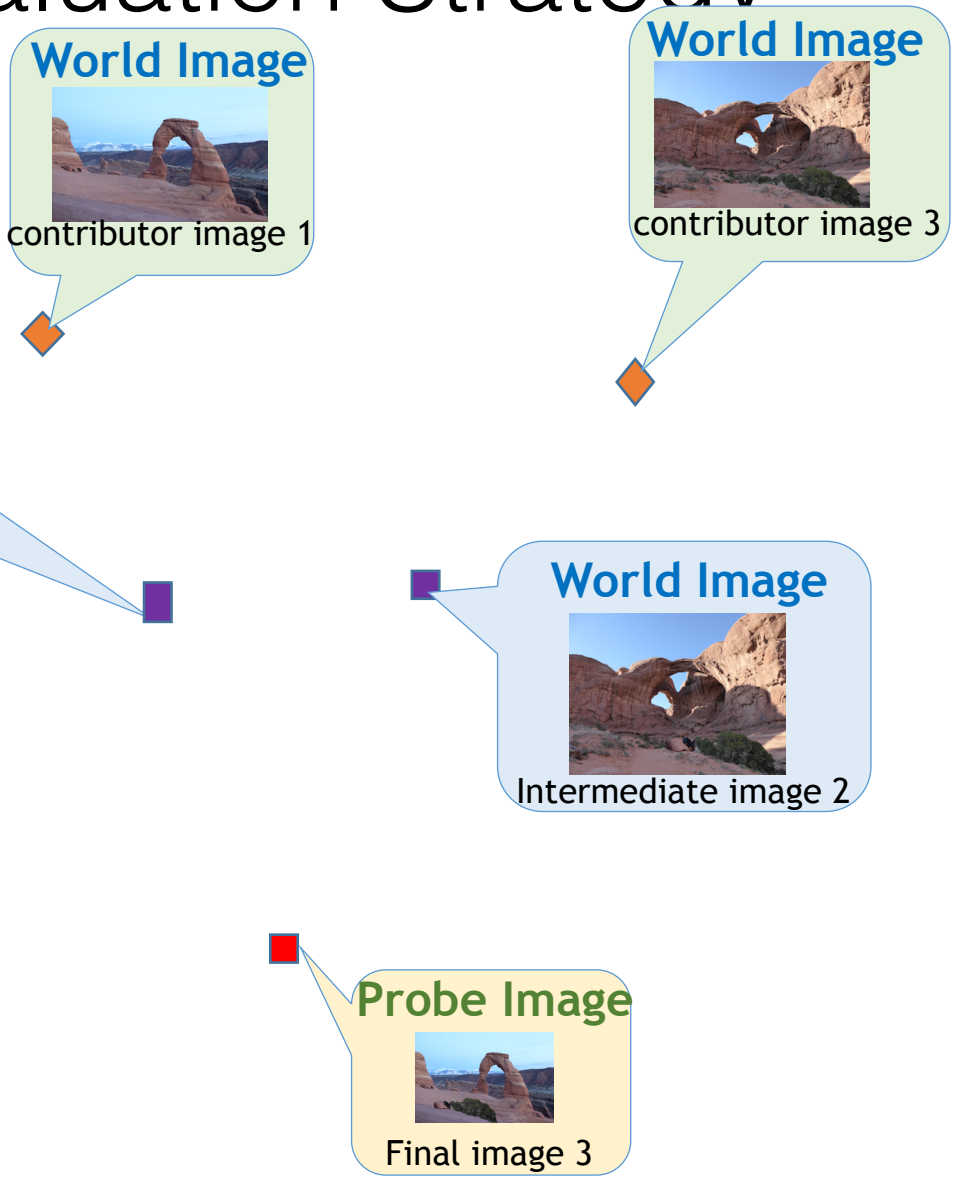
Step by Step Evaluation Strategy

- Step 1: Filtering



Step by Step Evaluation Strategy

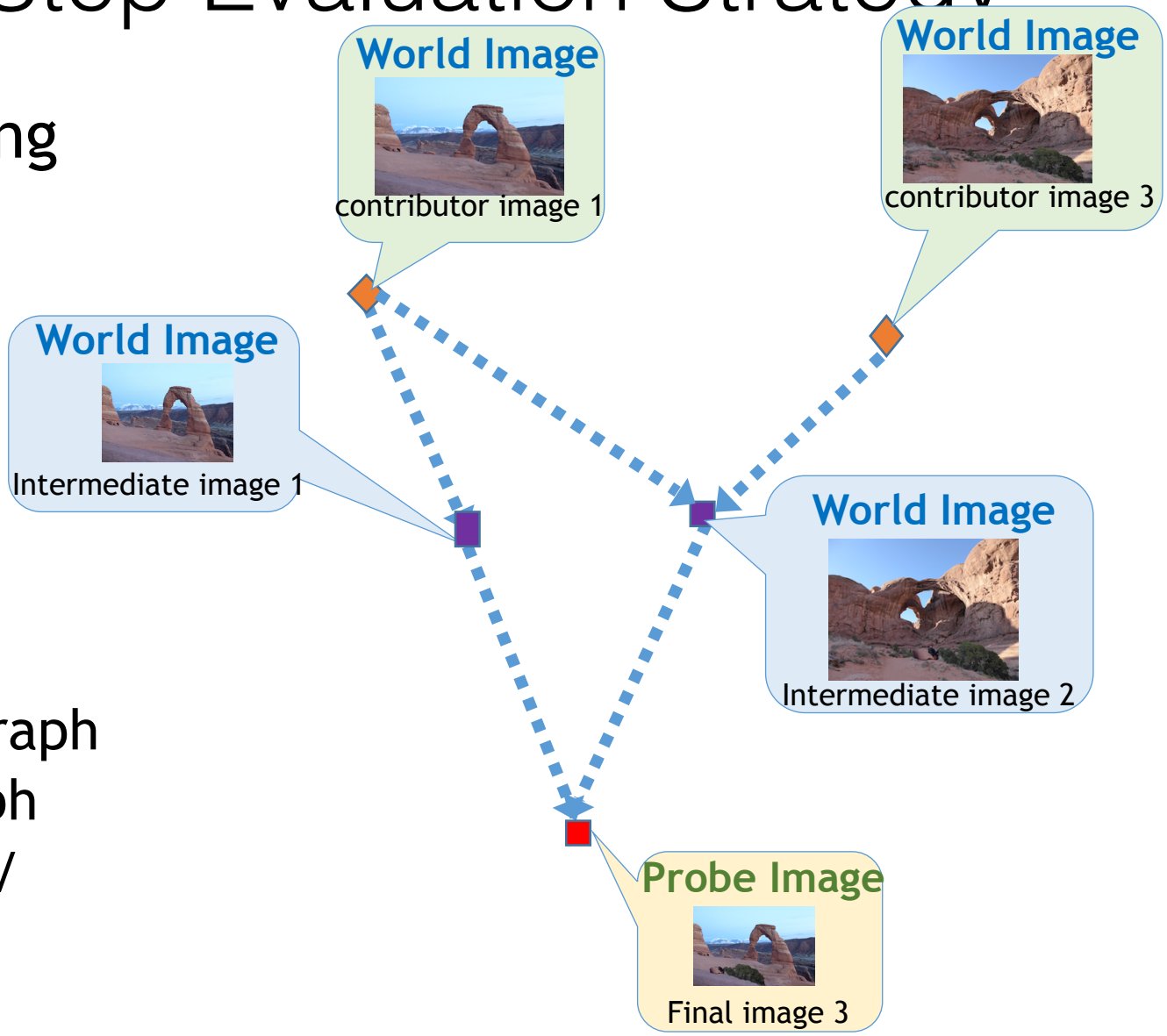
- Step 1: Filtering



- Step 2: Graph Building
  - undirected graph
  - directed graph
  - link relation / manipulation operation

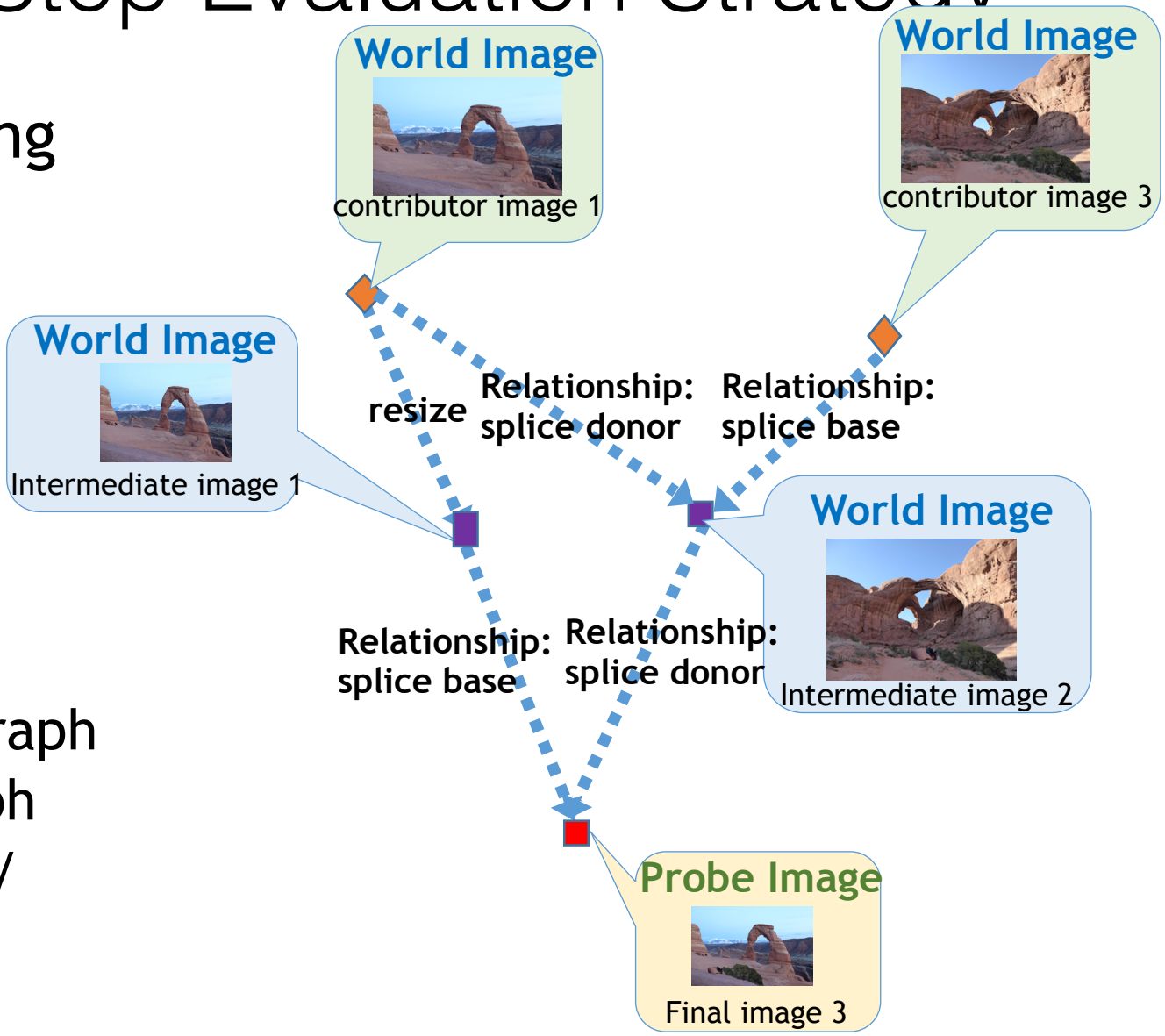
Step by Step Evaluation Strategy

- Step 1: Filtering
- Step 2: Graph Building
  - undirected graph
  - directed graph
  - link relation / manipulation operation

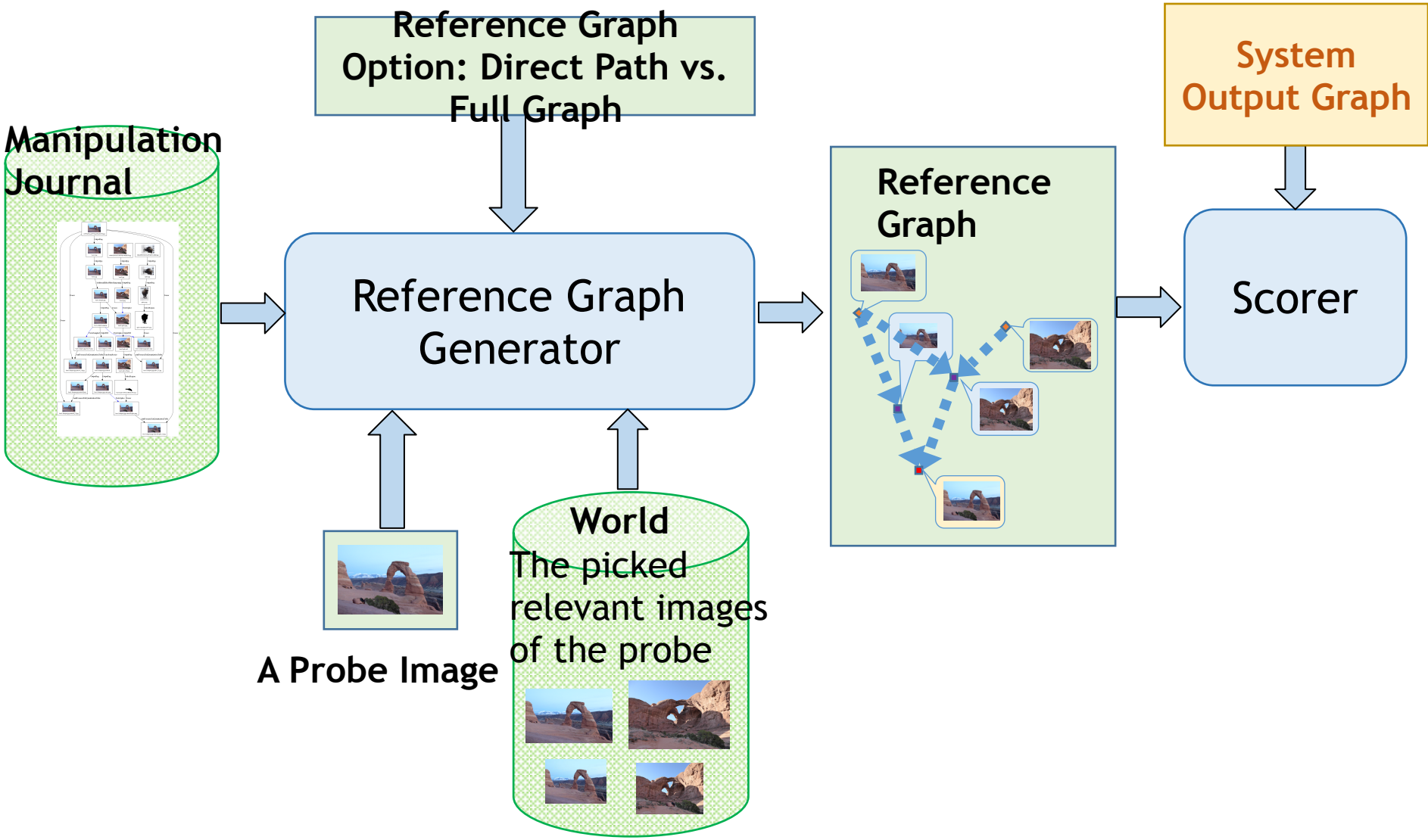


Step by Step Evaluation Strategy

- Step 1: Filtering
- Step 2: Graph Building
  - undirected graph
  - directed graph
  - link relation / manipulation operation



Provenance Evaluation Protocol



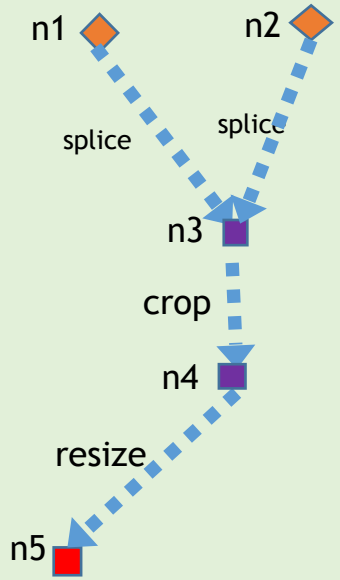
Reference Graph Generation

- Issues
  - The journal graph is manipulation-focused (more details):
    - Step-by-step description of the manipulation
  - Given limited info, there are ambiguities from an evaluation perspective (e.g. operation order etc.).
    - in real applications, the world dataset only contains a limited number of selected images
  - ❖Journal graph is not suitable to serve as the evaluation ground-truth (reference) graph directly.
- Approach
  - Generated the reference graph from journal graph based on a given probe and ancestors and descendants in the world data
  - Use the reference graph as the ground-truth graph for evaluation

Reference Graph Visualization

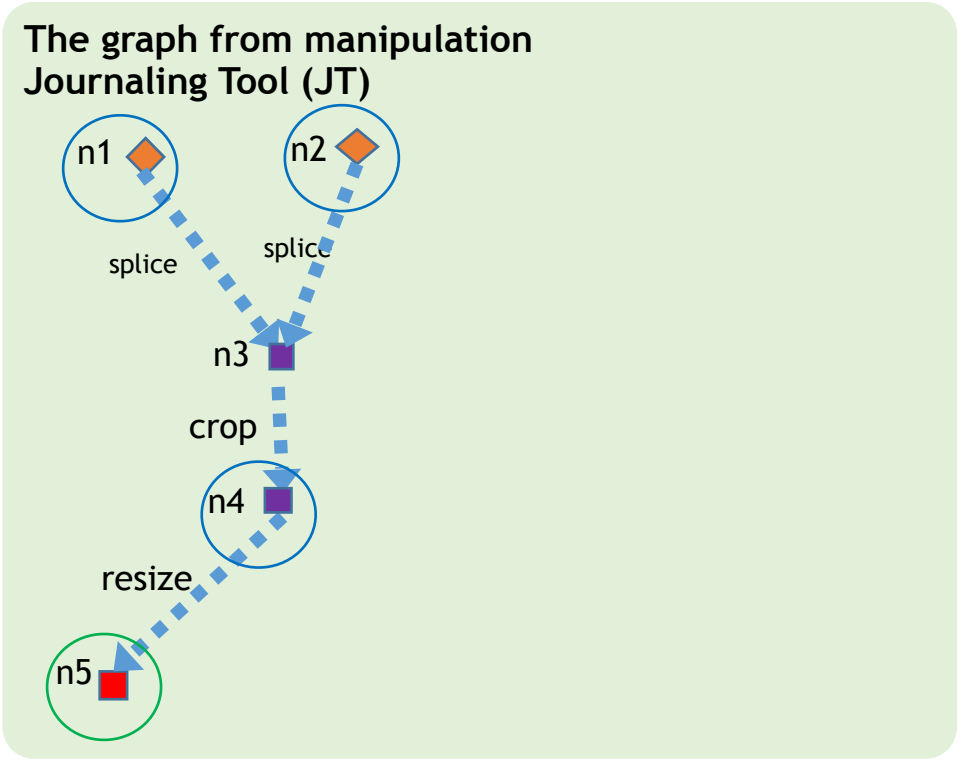
- Example: Given the probe (green) and world (blue) nodes.

The graph from manipulation Journaling Tool (JT)



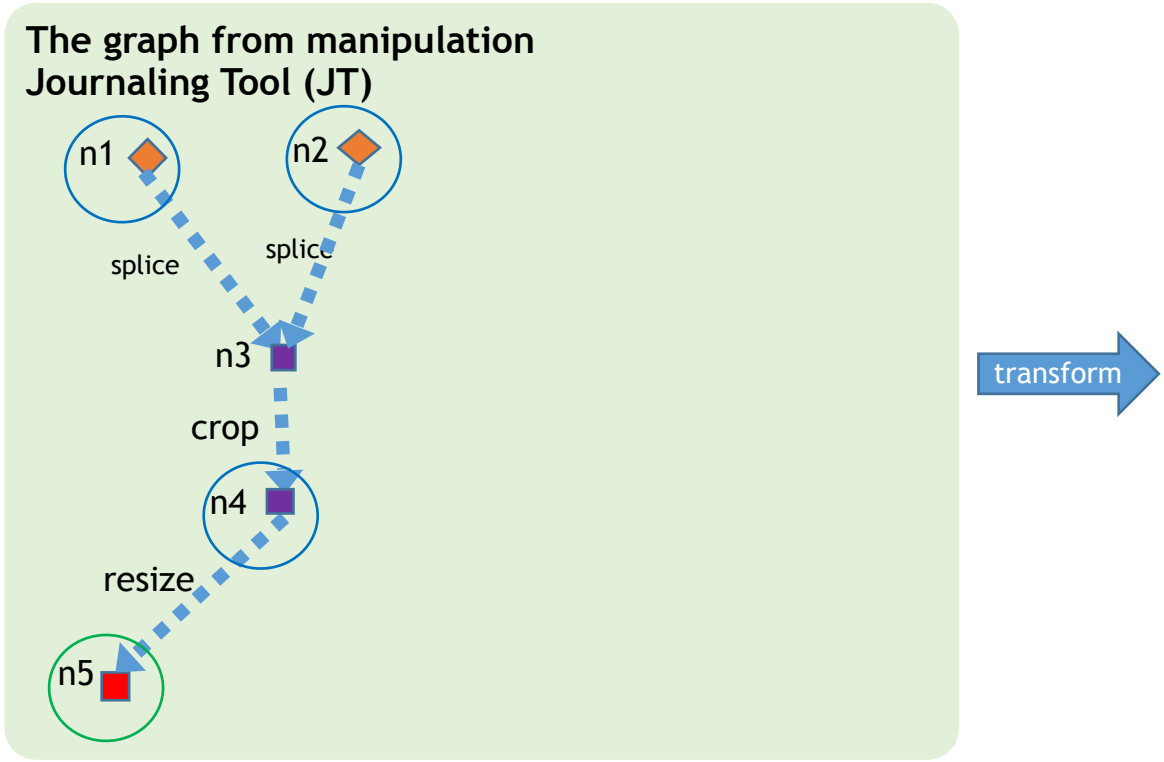
Reference Graph Visualization

- Example: Given the probe (green) and world (blue) nodes.



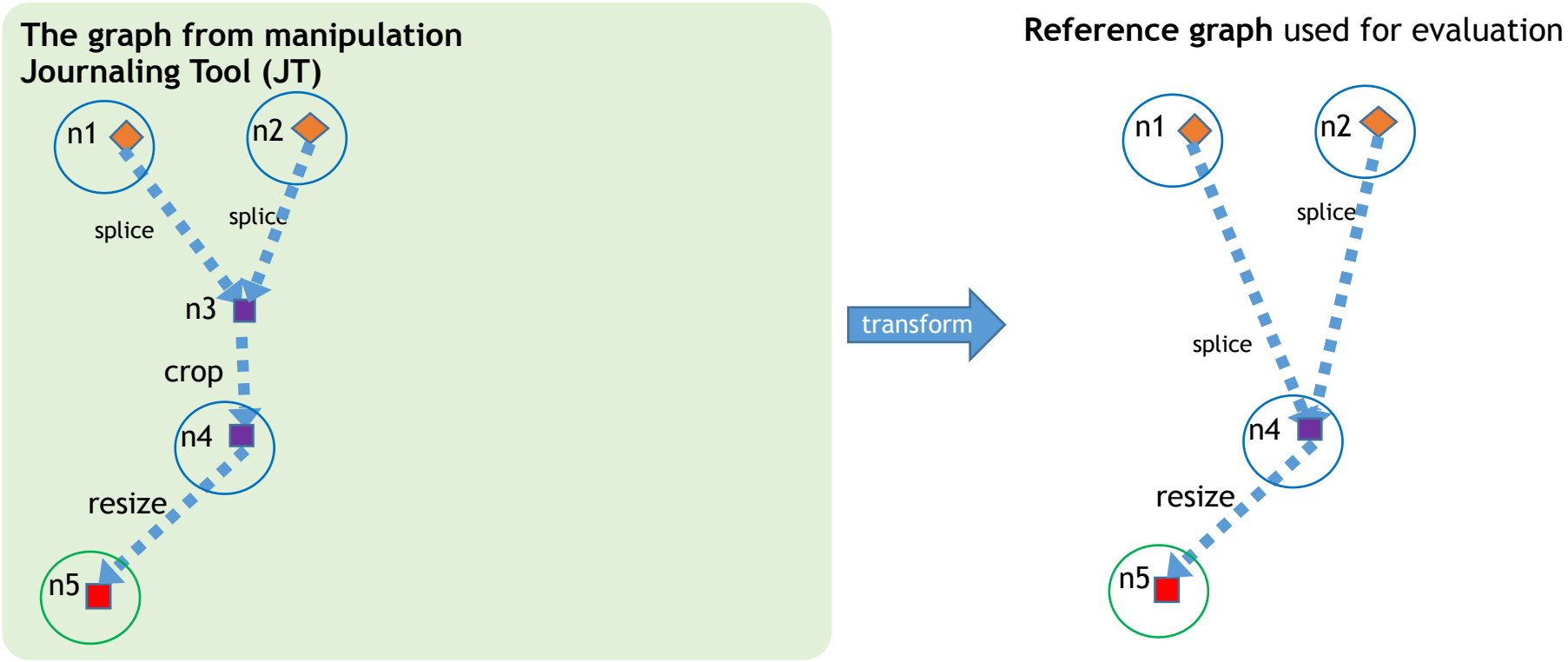
Reference Graph Visualization

- Example: Given the probe (green) and world (blue) nodes.



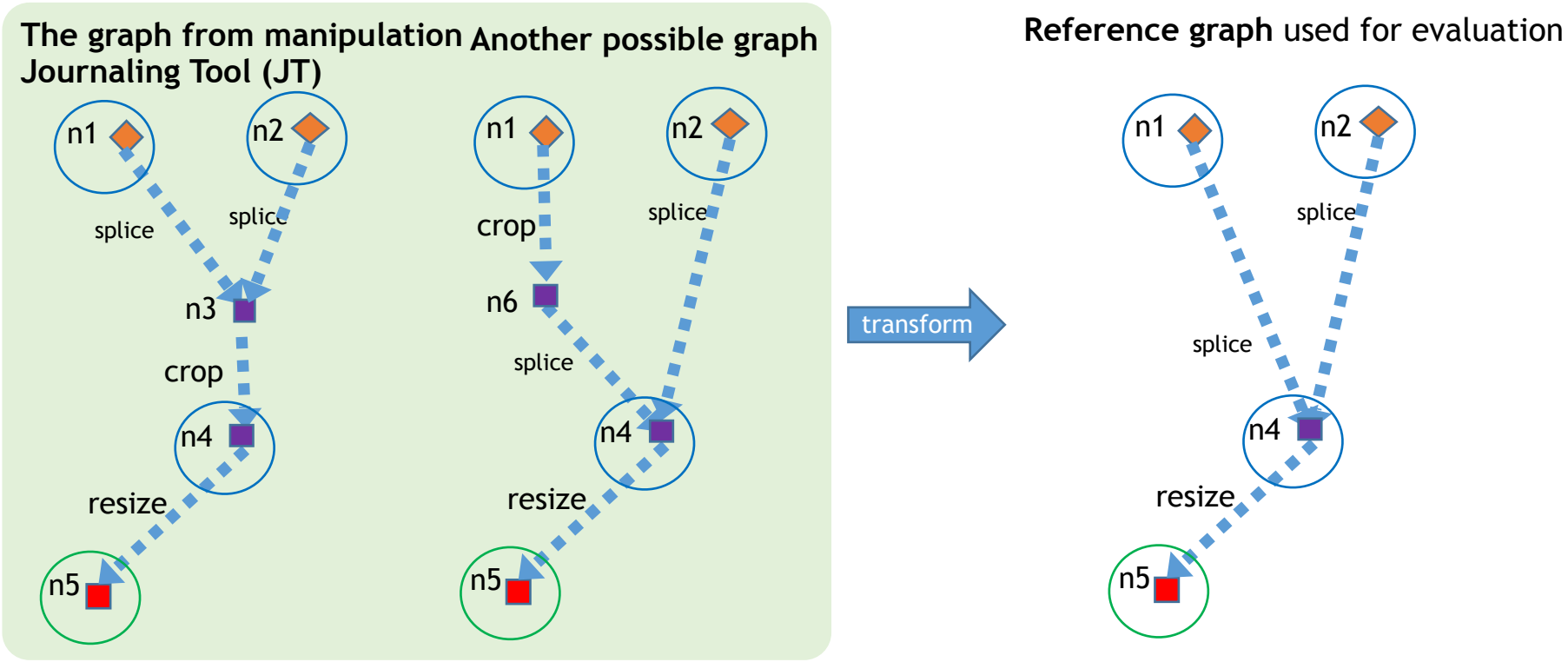
Reference Graph Visualization

- Example: Given the probe (green) and world (blue) nodes.



Reference Graph Visualization

- Example: Given the probe (green) and world (blue) nodes.



Reference Graph Building Example:

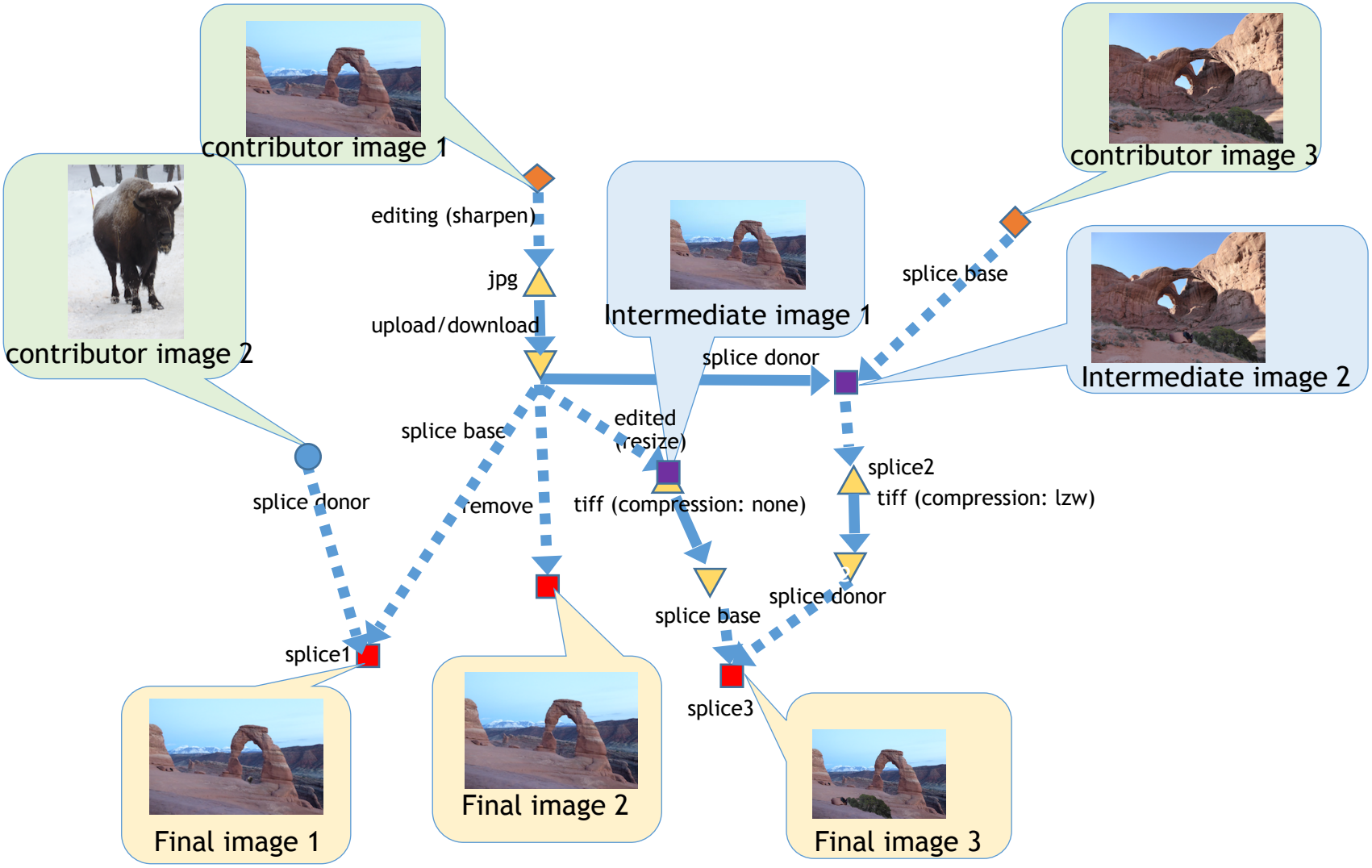
Initial Journal Graph

Reference Graph Building Example:

Initial Journal Graph

Reference Graph Building Example:

Concise Journal Graph



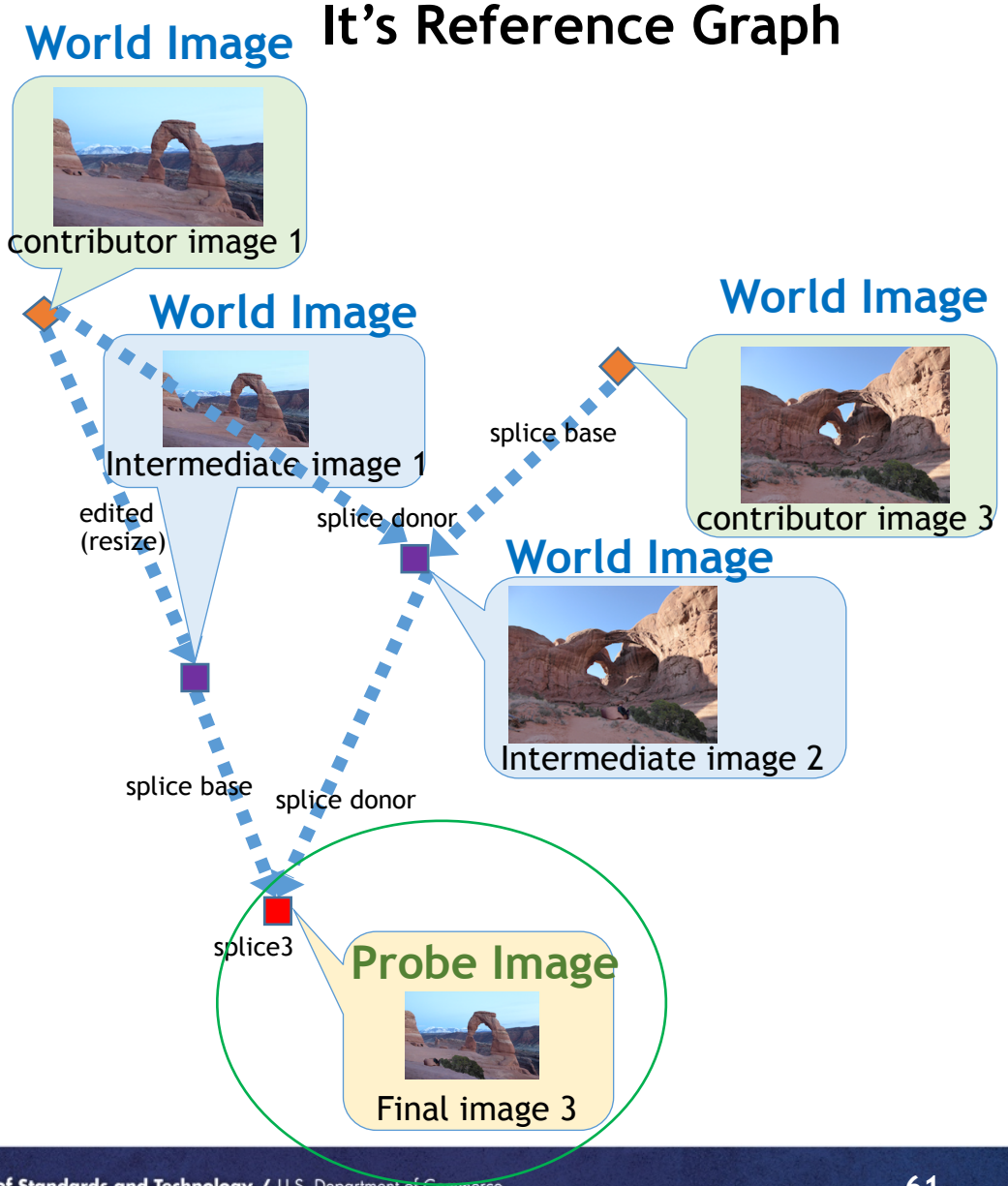
Reference Graph Building Example: Trial 1

System Input

Probe

World dataset

It's Reference Graph



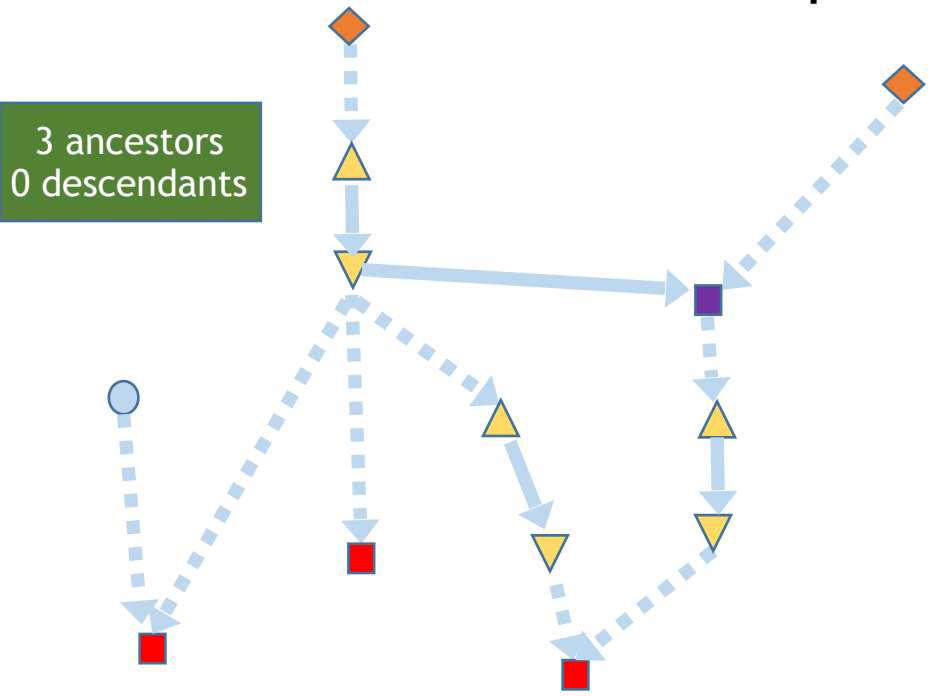
Options for Reference Graph:
Direct Path Limited vs. Full Graph

- Example 1:
  - probe: node with green circle;
  - world: all other nodes in concise graph

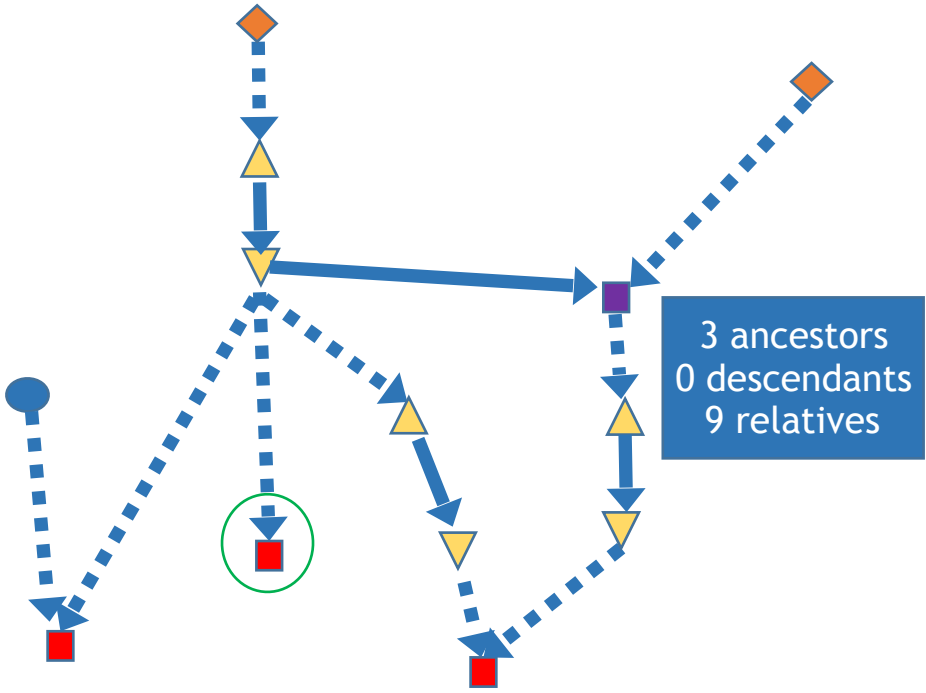
Direct Path Limited
All direct ancestors and decedents of a given probe

Full Graph
Recursively include all direct paths connected to the probe and all ancestors and decedents

Direct Path Reference Graph



Full Reference Graph



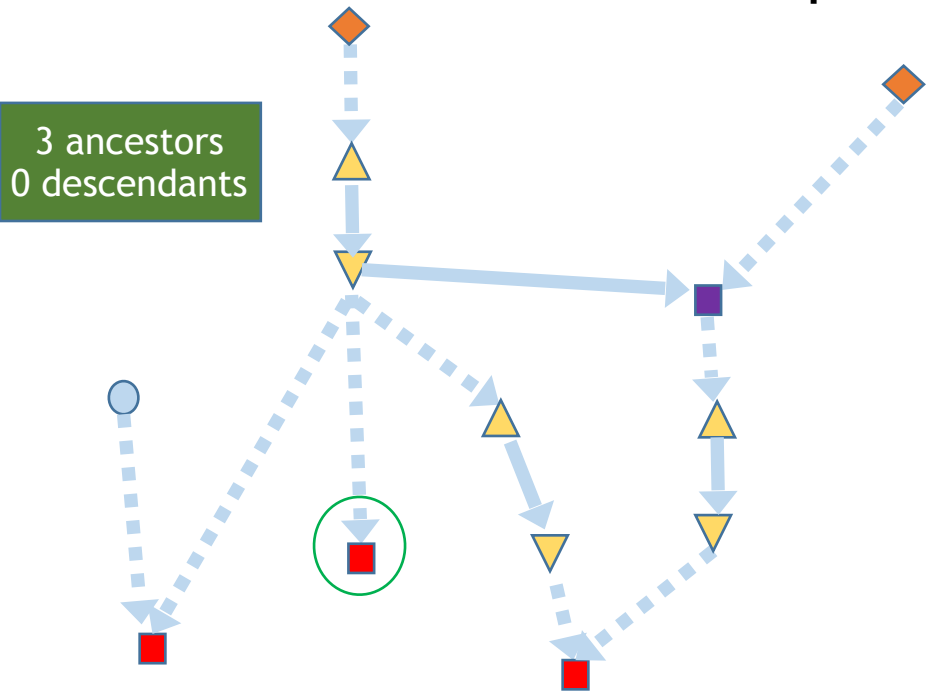
Options for Reference Graph:
Direct Path Limited vs. Full Graph

- Example 1:
  - probe: node with green circle;
  - world: all other nodes in concise graph

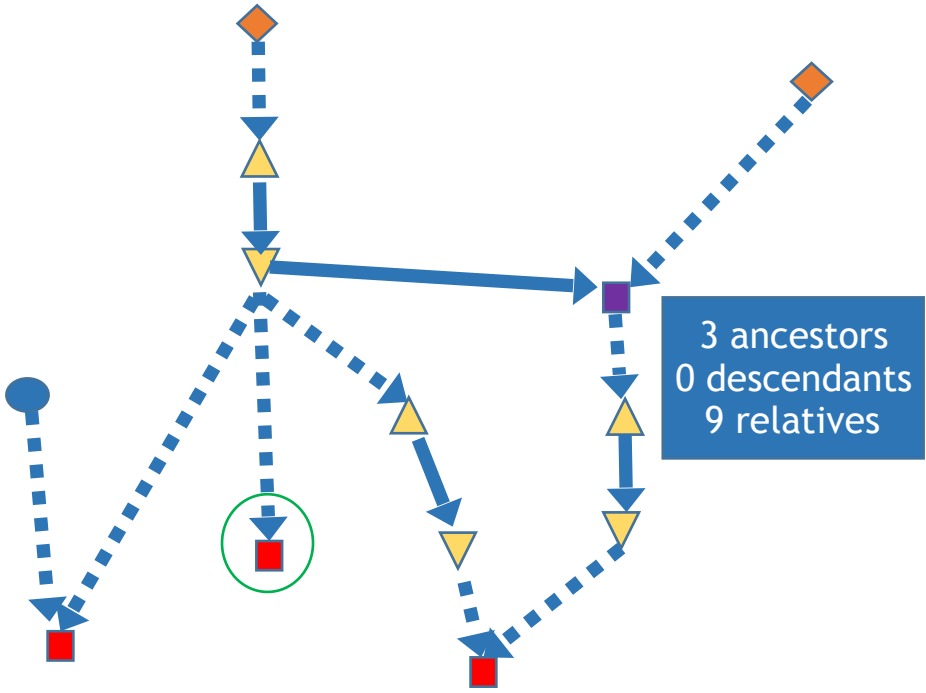
Direct Path Limited
All direct ancestors and decedents of a given probe

Full Graph
Recursively include all direct paths connected to the probe and all ancestors and decedents

Direct Path Reference Graph



Full Reference Graph



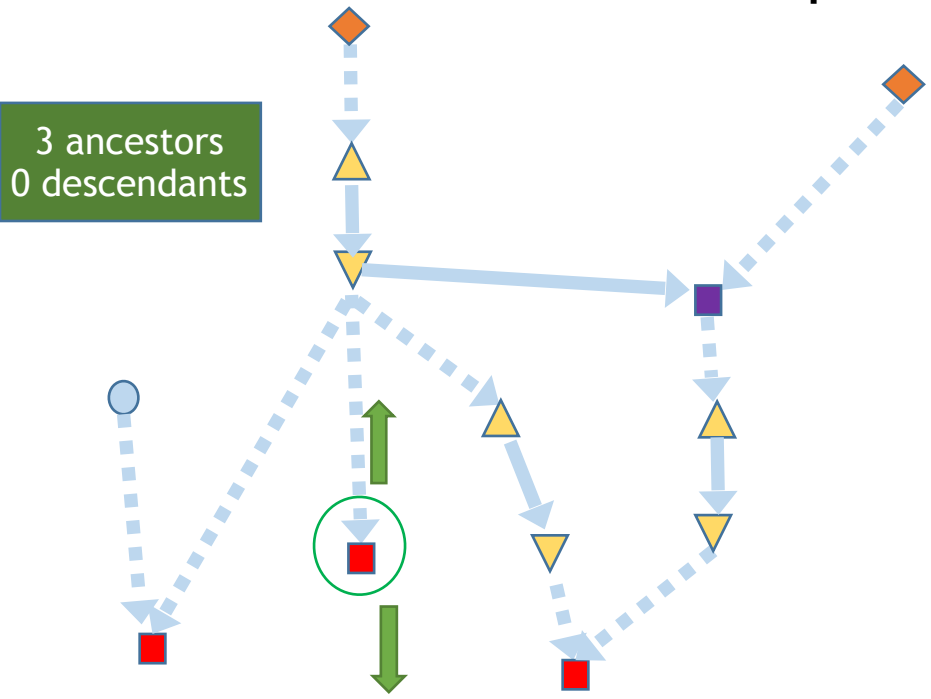
Options for Reference Graph:
Direct Path Limited vs. Full Graph

- Example 1:
  - probe: node with green circle;
  - world: all other nodes in concise graph

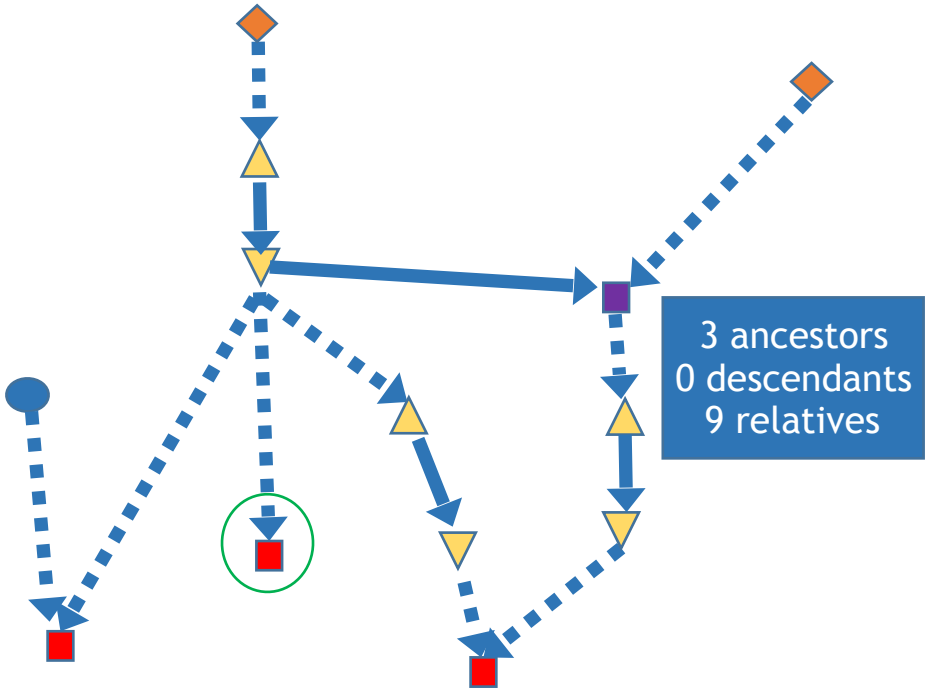
Direct Path Limited
All direct ancestors and decedents of a given probe

Full Graph
Recursively include all direct paths connected to the probe and all ancestors and decedents

Direct Path Reference Graph



Full Reference Graph



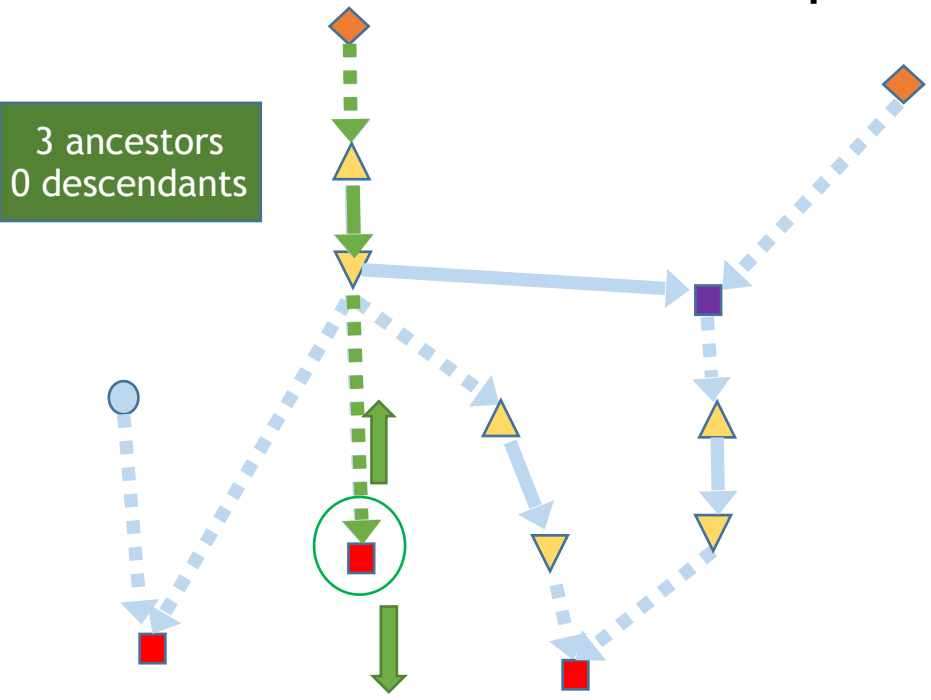
Options for Reference Graph:
Direct Path Limited vs. Full Graph

- Example 1:
  - probe: node with green circle;
  - world: all other nodes in concise graph

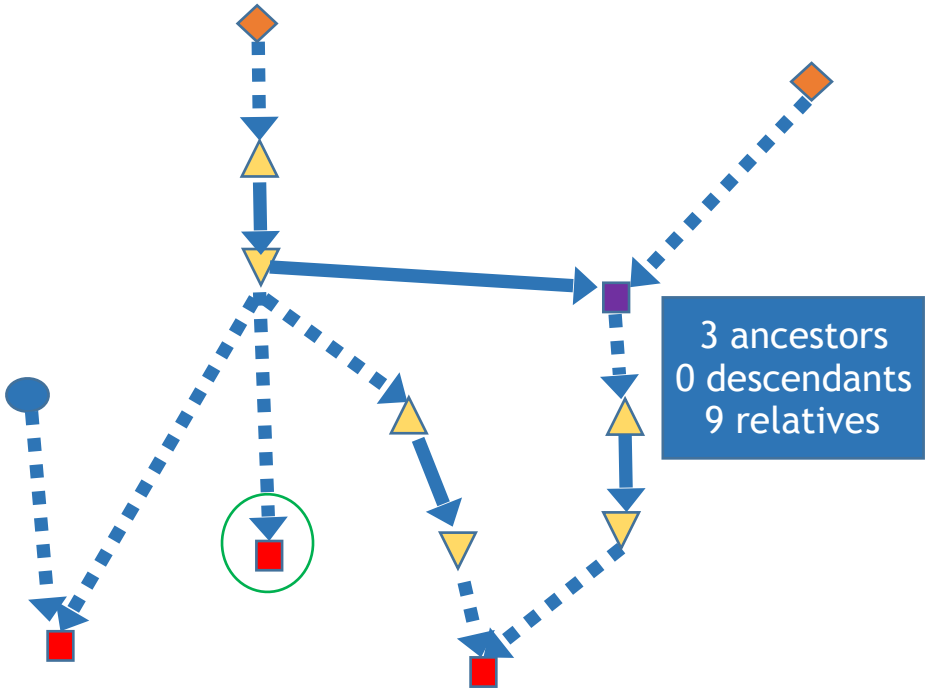
Direct Path Limited
All direct ancestors and decedents of a given probe

Full Graph
Recursively include all direct paths connected to the probe and all ancestors and decedents

Direct Path Reference Graph



Full Reference Graph



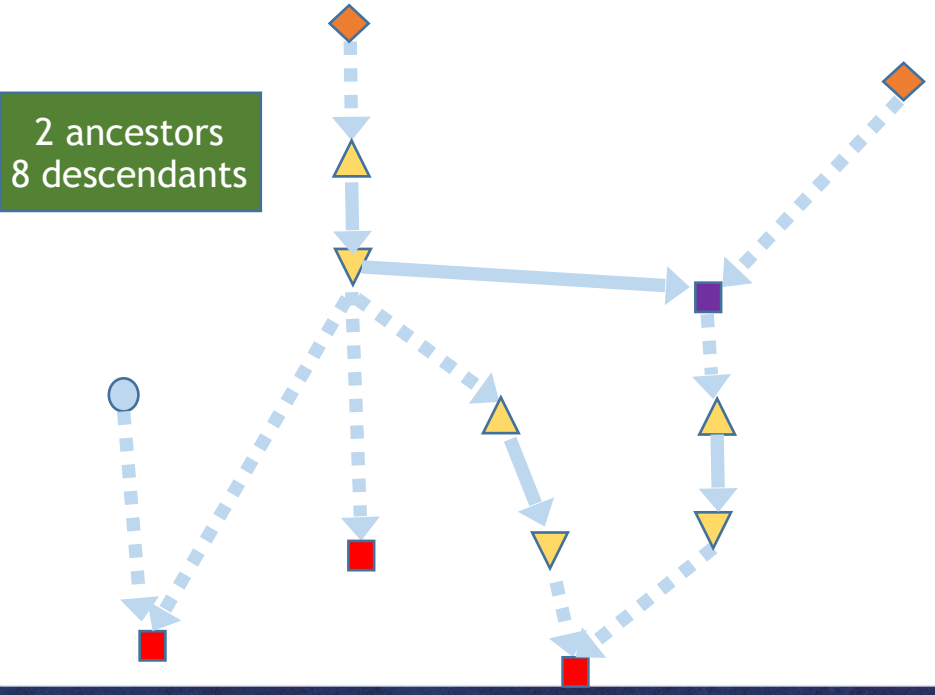
Options for Reference Graph:
Direct Path Limited vs. Full Graph

- Example 2:
  - probe: node with green circle;
  - world: all other nodes in concise graph

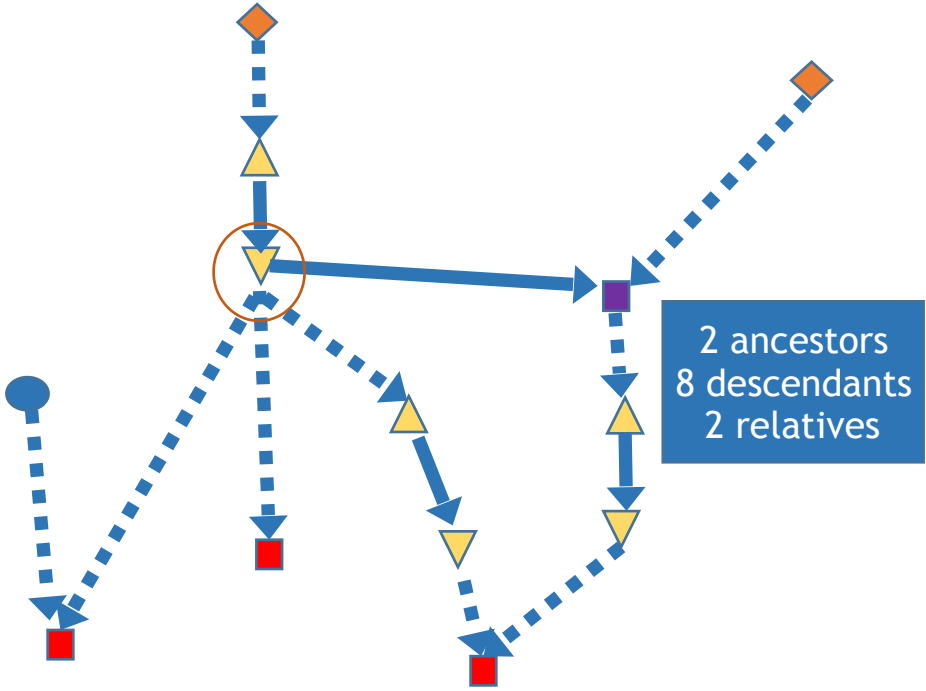
Direct Path Limited
All direct ancestors and decedents of a given probe

Full Graph
Recursively include all direct paths connected to the probe and all ancestors and decedents

Direct Path Reference Graph



Full Reference Graph



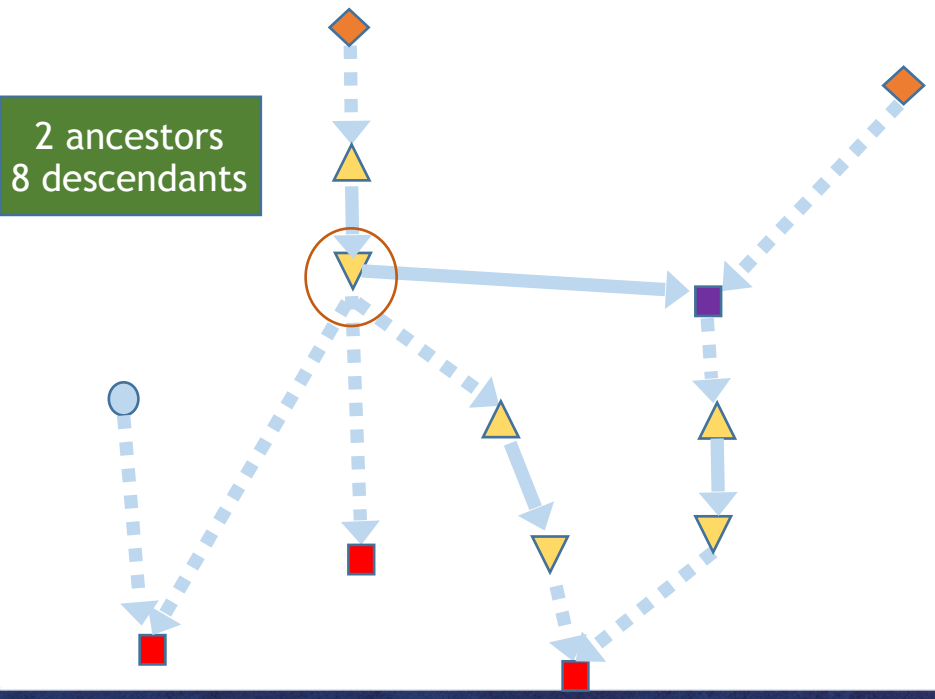
Options for Reference Graph:
Direct Path Limited vs. Full Graph

- Example 2:
  - probe: node with green circle;
  - world: all other nodes in concise graph

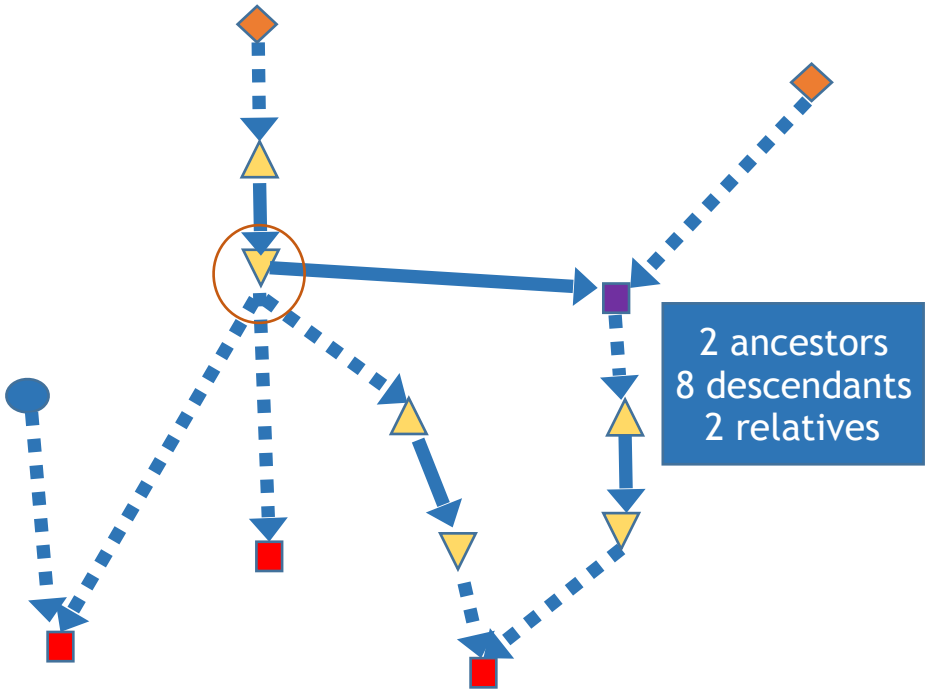
Direct Path Limited
All direct ancestors and decedents of a given probe

Full Graph
Recursively include all direct paths connected to the probe and all ancestors and decedents

Direct Path Reference Graph



Full Reference Graph



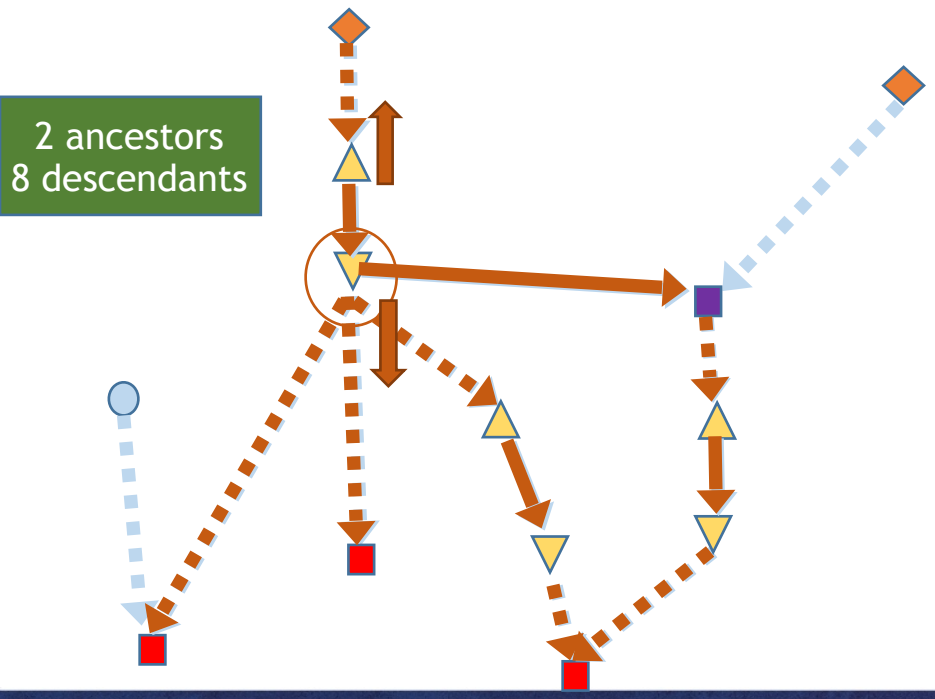
Options for Reference Graph:
Direct Path Limited vs. Full Graph

- Example 2:
  - probe: node with green circle;
  - world: all other nodes in concise graph

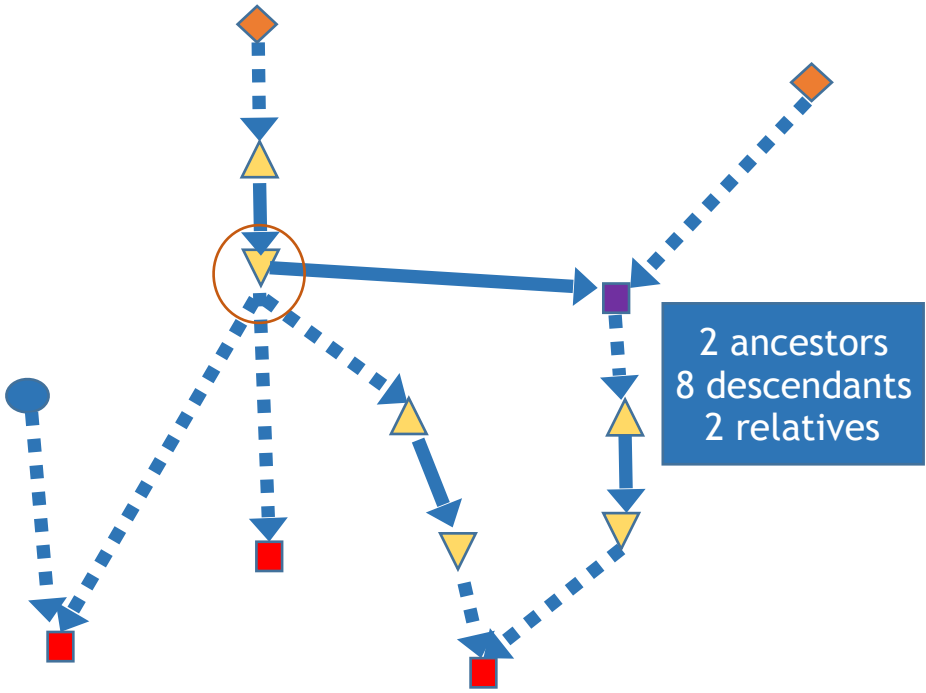
Direct Path Limited
All direct ancestors and decedents of a given probe

Full Graph
Recursively include all direct paths connected to the probe and all ancestors and decedents

Direct Path Reference Graph



Full Reference Graph



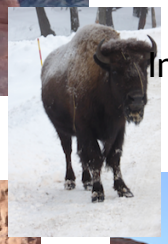
Reference Graph Building Example: Trial 2

System Input

Probe

Intermediate image 1

World dataset



Intermediate image 1

It's Reference Graph

contributor image 1

Intermediate image 1

resize

splice donor

splice base

contributor image 3

Intermediate image 2

splice base

splice donor

Final image 3

Reference Graph Building Example: Trial 2

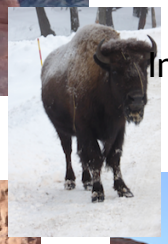
System Input

Probe

Intermediate image 1

World dataset

Intermediate image 1



It's Reference Graph

contributor image 1

Intermediate image 1

resize

splice donor

splice base

contributor image 3

Intermediate image 2

splice base

splice donor

Reference Graph Building Example: Trial 2

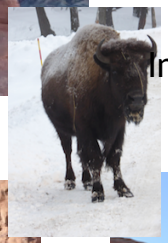
System Input

Probe

Intermediate image 1

World dataset

Intermediate image 1



It's Reference Graph

contributor image 1

Intermediate image 1

resize

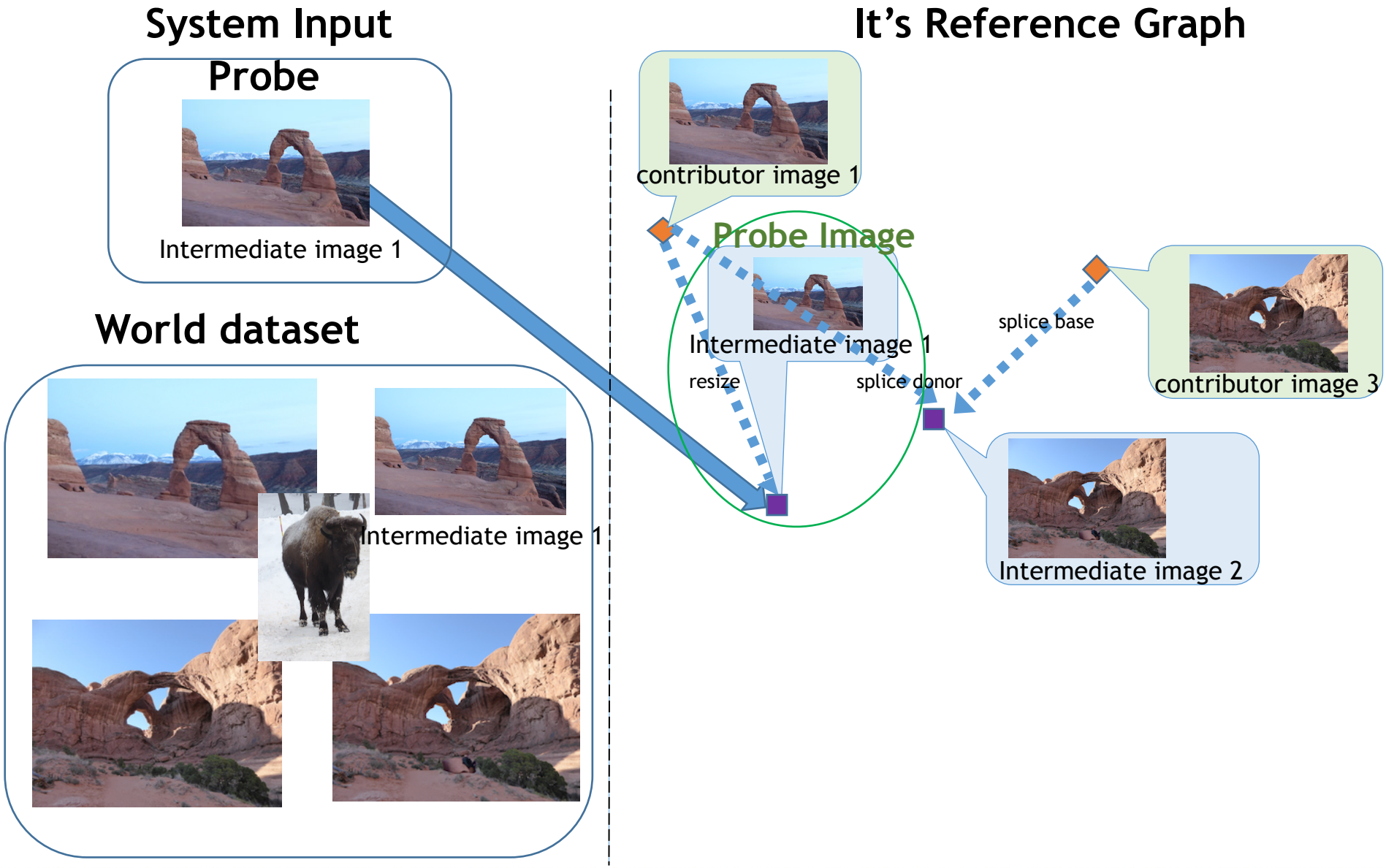
splice donor

splice base

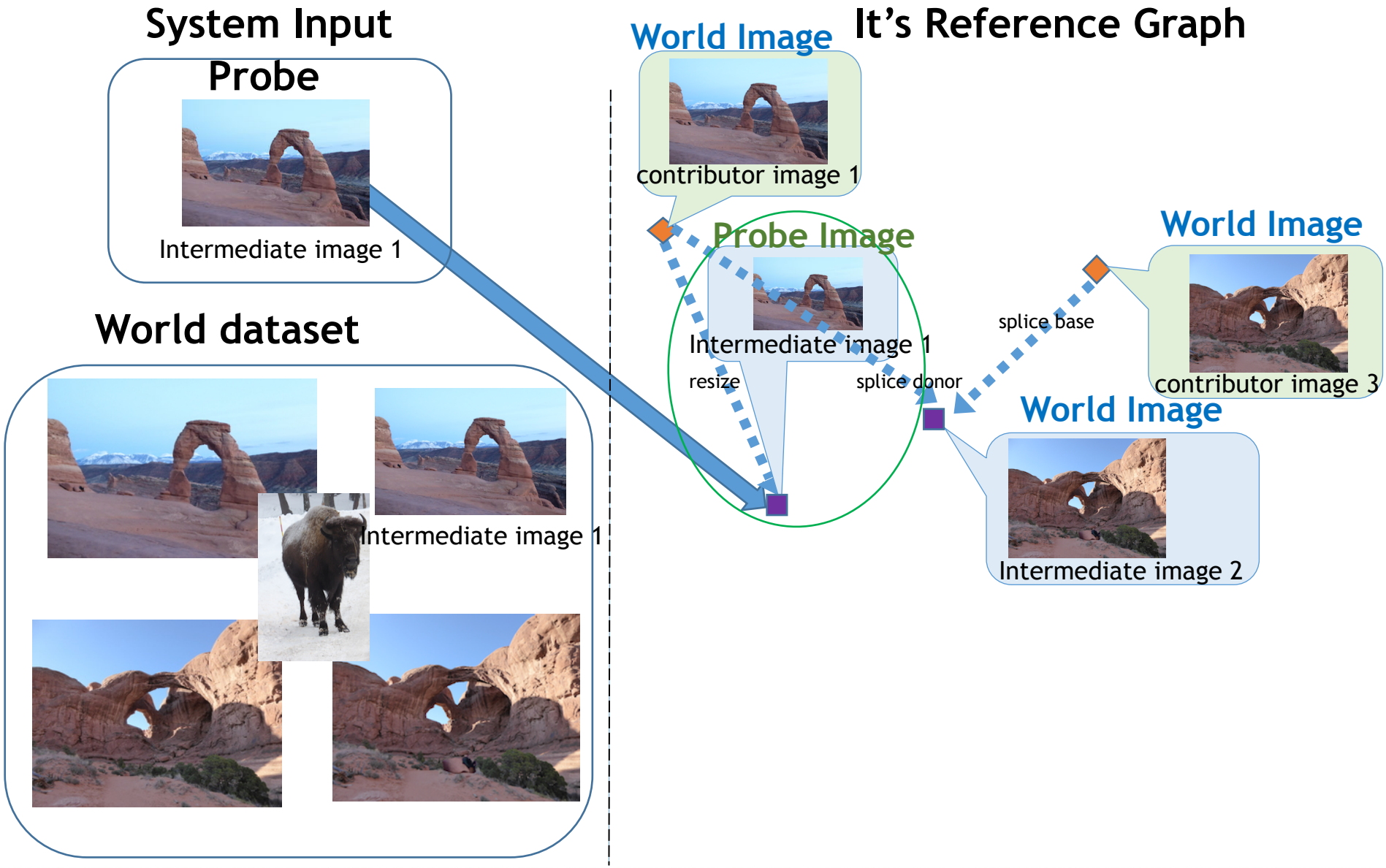
contributor image 3

Intermediate image 2

Reference Graph Building Example: Trial 2



Reference Graph Building Example: Trial 2



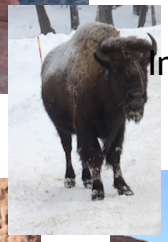
Reference Graph Building Example: Trial 2

System Input

Probe

Intermediate image 1

World dataset



Intermediate image 1

It's Reference Graph

World Image

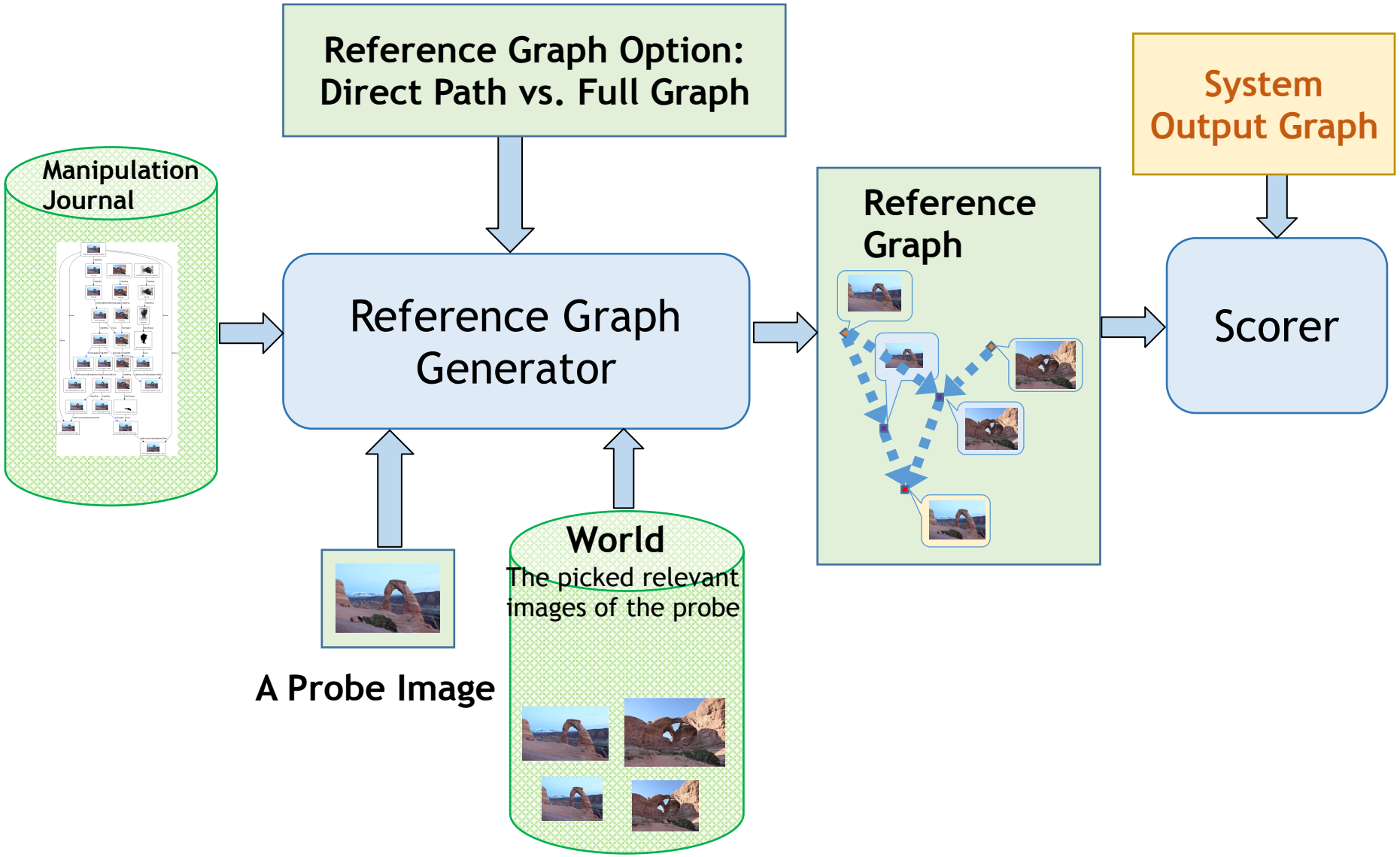
contributor image 1

Probe Image

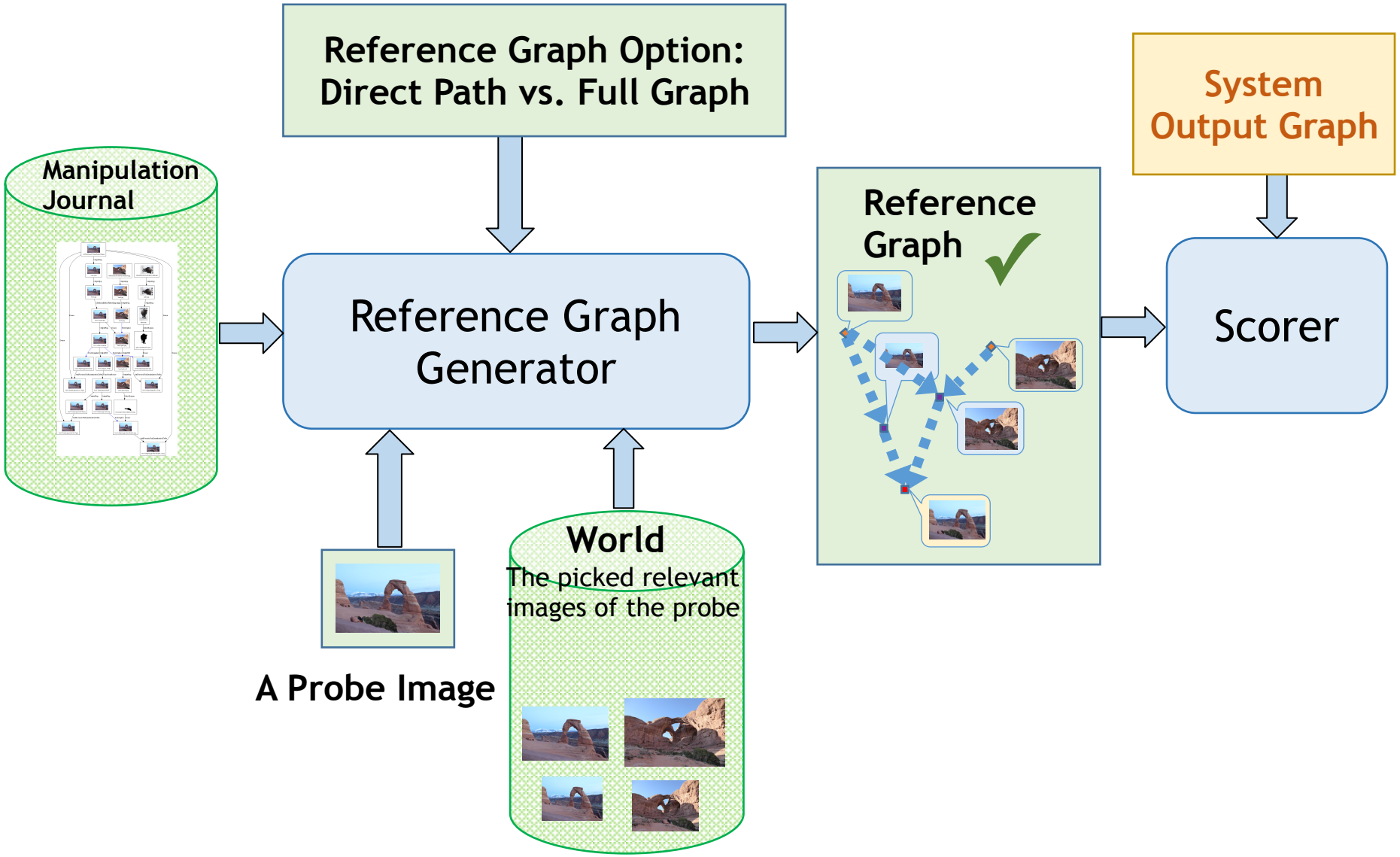
Intermediate image 1

resize

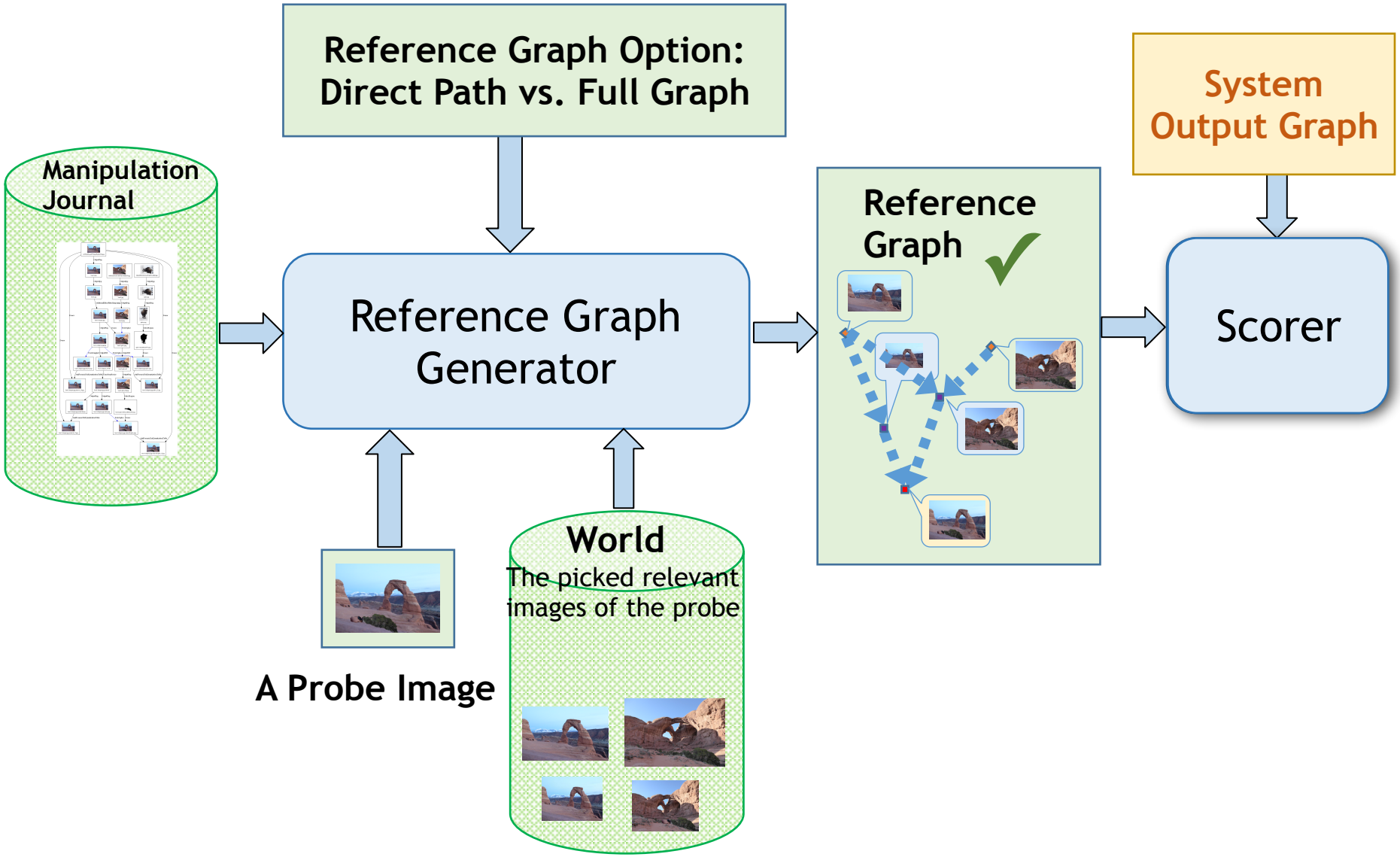
Provenance Evaluation Protocol



Provenance Evaluation Protocol



Provenance Evaluation Protocol



Provenance Filtering Task Evaluation Metrics

$$recall = \frac{|\{relevant\} \cap \{retrieved\}|}{|\{relevant\}|}$$

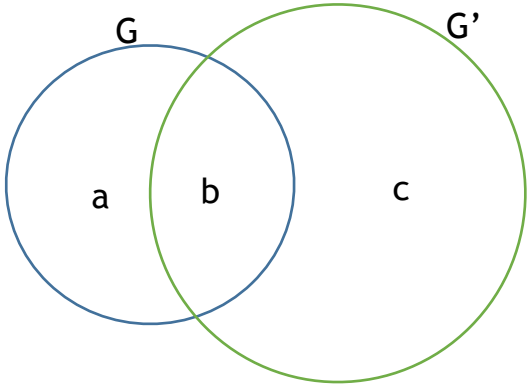
- The recall of first 100 images from the world dataset (≈1M) sorted by 'confidence score'
- Evaluated only true manipulated probes whose contributors are in the world data set
- Variations:
  - The depth of retrieval will be varied, e.g., recall@100, recall@50

Provenance Graph Building Task Evaluation Metrics: Overview

- Graph Similarity and Generalized F-measure
  - **Sim(nodes)**
    - scoring only on images
  - **Sim(links)**
    - scoring only on the relationship between images
    - link definition: correct direction and type
  - **Sim(nodes+links)**
    - scoring on both images and their relationships
- Customized metrics
  - The earliest source
- Cost function metrics
  - cost function approach: rule-based penalty
- Graph Edit Distance (suggested by Xu Zhang from Columbia team)

Similarity of **v**ertex **e**dge **o**verlap (Sim_{VEO}) [1]

$$Sim_{VEO}(G,G') = 2\frac{|V\cap V'|+|E\cap E'|}{|V|+|V'|+|E|+|E'|} = \frac{2b}{a+b+b+c}$$



$$recall = \frac{|\{relevant\}\cap\{retrieved\}|}{|\{relevant\}|} = \frac{b}{a+b}$$

$$precision = \frac{|\{relevant\}\cap\{retrieved\}|}{|\{retrieved\}|} = \frac{b}{b+c}$$

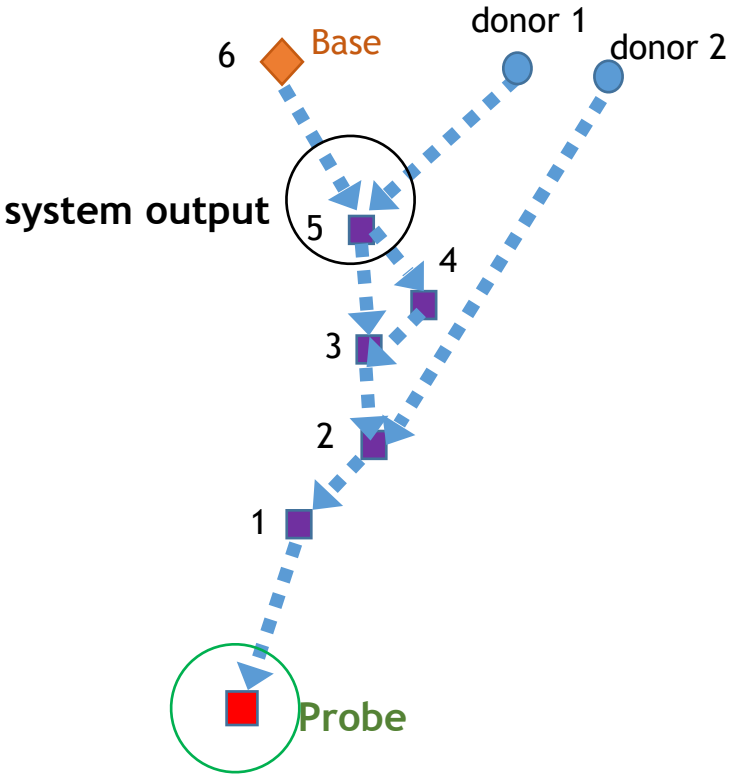
a = $|V| - |V \cap V'| + |E| - |E \cap E'|$
b= $|V \cap V'| + |E \cap E'|$
c = $|V'| - |V \cap V'| + |E'| - |E \cap E'|$

$$F = 2 * \frac{precision * recall}{precision+recall} = 2 * \frac{|\{relevant\}\cap\{retrieved\}|}{|\{retrieved\}|+|\{relevant\}|} = \frac{2b}{a+b+b+c} = Sim_{VEO}(G,G')$$

[1] P. Papadimitriou, A. Dasdan, and H. Garcia-Molina, "Web graph similarity for anomaly detection," *J. Internet Serv. Appl.*, vol. 1, no. 1, pp. 19–30, 2010.

Customized Metrics: The Earliest Source

- Idea:
  - Trace back from probe image to the base node, find the longest path including all related released intermediate nodes (use DAG topological sorting algorithm),
  - Assign each node in the path with a credit score = (the reversed order of this node)/(total links in the path).
- Example: CreditScore = 5/6
- Note: need further development
  - sum through all contributors?



Cost Function Based Metrics:

- Borrow idea from rule based penalty metrics [2]:
  - Compute penalty based on edges
    - A sum of DAG distance functions on the edges (i,j): $K_{ij}^{(p,q)}(G_1,G_2)$; p,q in range [0,1]. Assume q <= p.
    - If (i,j) is present in both: return 0.
    - If (i,j) is present in one but (j,i) is present in the other: return 1.
    - If (i,j) is present in one but not the other, return p.
    - If (i,j) is present in neither, return q.
      - Motivation: two complete DAG's have more in common than two empty DAG's.
  - Sum over all (unordered) pairs of distinct vertices.
- Customize weights(only idea, need further development):
  - direction wrong: less penalty, r
  - edge in ground-truth, not in system output, p
  - edge in system output, not in ground-truth, q

[2] E. Malmi, N. Tatti, and A. Gionis, "Beyond rankings: comparing directed acyclic graphs," Data Min. Knowl. Discov., vol. 29, no. 5, pp. 1233–1257, Sep. 2015.

Graph Edit Distance (GED)

$$GED(g_1, g_2) = \min_{(e_1,\ldots,e_k)\in\mathcal{P}(g_1,g_2)} \sum_{i=1}^{k} c(e_i)$$

- Suggested by Xu Zhang
- The set of elementary graph edit operators typically includes:
  - vertex insertion
  - vertex deletion
  - vertex substitution
  - edge insertion
  - edge deletion
  - edge substitution
- NP-hard

Provenance Tasks' System output format

- One unified file format to handle both provenance tasks:
  - Provenance filtering system output is a subset of the full output file.
- Performer's system output for each probe image:
  - Directed Acyclic Graph (DAG) represented by a json file.
  - Each node represents an image with confidence score for filtering task.
  - Each link represents a directional relationship between two images.
    - Optional: the link may contain a field for another confidence score of the relationship (for graph building task).
    - links omitted for provenance filtering task.
  - Note:
    - Could contains multi- connected components since not all links may be discovered by the system
    - Must be DAG (topological order/sort algorithm for DAG validation)
    - provenance filtering task: 200 nodes.

Thank You for Your Attention!

NIST Medifor Team: medifor-nist@nist.gov