Open Media Forensics Challenge 2022 Evaluation Plan

Date: 2022-03-03

Haiying Guan, Yooyoung Lee, and Lukas Diduch

NIST Open Media Forensics Challenge Team

NIST

Table of Contents

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

This document describes the ongoing Open Media Forensics Challenge (OpenMFC) 2022 evaluation. The following sections will provide all information necessary for a user or group to successfully participate in the evaluation.

The NIST OpenMFC is an open evaluation organized by the National Institute of Standards and Technology (NIST) to promote technologies that can automatically detect and locate manipulations/deepfakes in imagery (i.e., images and videos). It is a continuation of the NIST Media Forensics Challenge (MFC) effort supporting the DARPA MediFor Program (2017-2020).

The OpenMFC challenge provides a set of data (e.g., training, validation, test sets) to participants to train and run a system on their own hardware platform and submit their system outputs to a web-based leaderboard for displaying scoring results.

OpenMFC related questions can be emailed to mfc_poc@nist.gov.

2 TASKS AND CONDITIONS

The primary goal of the challenges is to engage the public research community worldwide to work on the latest media forensics topics. The OpenMFC focuses on three major tasks:

1) Manipulation Detection
2) Deepfakes Detection
3) Steganography Detection

2.1 TASKS

In this evaluation media is described in the following way: A 'base' indicates original media with high provenance, while a 'probe' indicates a test media. A 'donor' indicates another media whose content was donated into the base media and generated the probe media.

2.1.1 MANIPULATION DETECTION

For the Manipulation Detection (MD) task, the objective is to detect if a probe has been manipulated, and if so, to spatially localize the edits. Manipulation in this context is defined as deliberate modifications of media (e.g., splicing and cloning etc.) and localization is encouraged but not required for OpenMFC.

The MD task includes three subtasks, namely, image manipulation detection (IMD), image splice manipulation detection (ISMD), and video manipulation detection (VMD). The IMD subtask will detect if a probe image has been manipulated. To support entry-level public participants, we add a new subtask, ISMD, in the OpenMFC 2022. The ISMD subtask will detect if a probe image has been spliced. The VMD subtask will detect if a probe video has been manipulated.

For each MD trial, which consists of a single probe, the MD system must render a confidence score (see Section 5.2) with higher numbers indicating the probe is more likely to have been manipulated. The primary metric for detection performance will be Area Under the Receiver Operating Characteristic (ROC) Curve (AUC) (see Section 6.1.2).

For the localization task, the system output mask must be relative to the input probe and indicate the confidence of each location was manipulated. The form of the system-provided masks is defined in Appendix D-b.

2.1.2 DEEPFAKES DETECTION

The deepfakes detection (DD) task will detect if a probe has been manipulated using Generative Adversarial Network (GAN) or Deepfake based techniques. The DD task includes two subtasks, namely, image deepfakes detection (IDD) and video deepfakes detection (VDD). The IDD subtask will detect if a probe image has been manipulated based on GAN models while the VDD subtask will detect if a probe video has been manipulated based on Deepfake models.

For each DD trial, which consists of a single probe, the DD system must render a confidence score with higher numbers indicating the probe image is more likely to have been manipulated using GAN-based techniques. The primary metric for measuring detection performance will be Area Under the Receiver Operating Characteristic (ROC) Curve (AUC) (see Section 6.1.2).

2.1.3 STEGANOGRAPHY DETECTION

The Steganography Detection (StegD) task will detect if a probe is a stego image, which contains the hidden message either in pixel values or in optimally selected coefficients [1].

For each StegD trial, which consists of a single probe image, the StegD system must render a confidence score with higher numbers indicating the probe image is more likely to be a stego image. The primary metric for measuring detection performance will be Area Under the Receiver Operating Characteristic (ROC) Curve (AUC) (see Section 6.1.2).

2.2 PROTOCOL

All probes must be processed independently of each other within a given task and across all tasks, meaning content extracted from probe data must not be used to inform or affect another probe (or another task).

The OpenMFC evaluation does not provide training sets for any tasks. Previously released development data and evaluation data for the MD task is not designed to be used for training purposes on future evaluation cycles. Developers can use this data for training sets at their own risk.

All machine learning or statistical analysis algorithms must complete training, model selection, and tuning algorithm parameters prior to running the OpenMFC test data.

3 DATA RESOURCES

Each OpenMFC data set makes use of three main directories: 'probe', 'indexes', and 'reference'. They are explained in the following sections.

3.1 PROBE DIRECTORY

The OpenMFC 'probe' directory contains images or videos that will be forensically analyzed. The images or videos may be either manipulated or non-manipulated. The OpenMFC, images and videos may be of any format of camera output.

Table 1 summarizes the approximate number of probes of the OpenMFC datasets.

Task	Subtask	Dataset	Approximate number of probes
MD	IMD	OpenMFC20_Image_MD	16 000
	ISMD	OpenMFC22_ImageSplice_MD	2 000
	VMD	OpenMFC20_Video_MD	1 530
DD	IDD	OpenMFC20_Image_DD	1 340
	VDD	OpenMFC20_Video_DD	118
	VDD	OpenMFC22_Video_DD	TBD
StegD	StegD	OpenMFC22_Image_StegD	480

3.2   INDEXES DIRECTORY

The OpenMFC 'indexes' directory contains a system index file for each task. An index file is a comma-separated value (CSV) file which lists the images or videos a system must process (see Section 4.1 and Appendix B for details).

3.3   REFERENCE DIRECTORY

The OpenMFC 'reference' directory contains a subdirectory for each evaluation subtask, i.e., IMD, ISMD, VMD, IDD, VDD, and StegD. In each of these detection directories there will be two types of data: (1) the reference files that contain the "ground-truth" and metadata about trial probes and (2) a subdirectory containing the reference masks.

3.3.1   REFERENCE FILES

Three files constitute the reference files for the detection tasks. The main reference file follows the naming convention <DataSet_Name>-<TaskID>-ref.csv. This file contains seven columns that describe each trial. These columns are described below. Additional columns that are documented in the data release will be used for analysis.

TaskID	The type of system output, e.g., "manipulation"
ProbeFileID	The ID of the probe, e.g., 001f9af3165a39c9e42aee922f874326
ProbeFileName	The partial path name to the probe file (relative to the top node of the data distribution), e.g., probe/001f9af3165a39c9e42aee922f874326.jpg
IsTarget	Boolean indicating if the probe is a manipulated image, i.e., "Y" | "N"
ProbeMaskFileName	The partial path name to the manipulation mask for the probe (relative to the top node of the data distribution), e.g., reference/splice/mask/001f9af3165a39c9e42aee922f874326-mask.png if IsTarget = "Y", blank otherwise (i.e., no content).
BaseFileName	The partial path name to the base image within the world data set (relative to the top node of the data distribution), e.g., world/ 02264e9cb30bf0e89020b7c283e3aad6.tif if IsTarget = "Y", blank otherwise.
JournalName	The name of the manipulation journal for which the probe was extracted, e.g., oof7oxgiqjprd4ou4lq75wtnvdlmwhkk if IsTarget = "Y", blank otherwise.

For the video tasks, there are two additional columns:

FrameCount	The total number of frames in the probe, e.g., 1200
FrameRate	The number of frames displayed per second in the probe, e.g., 24

For each probe for a given detection TaskID, the file <DataSet_Name>-<TaskID>-ref-probejournaljoin.csv documents the journal from which the probe came as well as the operation(s), identified by the before-operation-node and after-operation-node, referenced in the <DataSet_Name>-<TaskID>-ref-journalmask.csv.

Journals may include sub-graphs that do not apply to a given probe; only entries that pertain to a given probe are in the probejournaljoin file.

ProbeFileID	Same as above
JournalName	Same as above
StartNodeID	The starting media ID within the journal whose operation is included in the probe, e.g., if77i8v5clk3g2btmpz038hhrnx499s3-TGT-01
EndNodeID	The ending media ID within the journal whose operation is included in the probe, e.g., if77i8v5clk3g2btmpz038hhrnx499s3-TGT-02-FILL
BitPlane	The bit position of the manipulation in the JPEG2000 mask, e.g., $BP = 2$; the (i,j)-th pixel was modified by the operation if $1 = pixel_{i,j} \wedge 2^{BP-1}$

For the video tasks, there is one additional column:

VideoTaskDesignation	Defines which video task can be supported by the probe, i.e., "spatial", "spatial-temporal", or "temporal"

The file <DataSet_Name>-<TaskID>-ref-journalmask.csv documents all masks for each operation in the journal, including manipulations not necessarily included in the probe. Each row is an operation; for localizable operations, a color is provided.

JournalName	Same as above
StartNodeID	Same as above
EndNodeID	Same as above
Operation	The manipulation operation type from the journal JSON file, e.g., "PasteSplice"
Color	The RGB color as a triplet of integers between 0 and 255, e.g., 255 10 0
Purpose	The semantic purpose of the manipulation, e.g., an object "remove" can be accomplished with several types of operations
OperationArgument	Arguments supplied with the given operation, e.g., "natural object"
AutoDesignName	The design operation type applied to generate output of auto manipulations, e.g., LaunderingFacebook_Option_1
PlugInName	The name of the specific plug-in used for auto manipulations, e.g., "GaussianLocalBlur"
PlugInArguments	The arguments used for the plug-in, e.g., "kernelsize 9"
SubCategory	Additional information in regard to the operation, e.g., GAN

For videos, there are 5 additional columns.

VideoTime	The time segments, given as millisecond intervals, where video manipulation occurs (each video starts at 0 ms), e.g., $[[0, 600], [1000, 7000], [1100, 1200]]$
VideoFrame	The frame segments, given as integer intervals, where video manipulation occurs (each video starts at Frame 1), e.g., $[[1, 20], [40, 230], [37, 40]]$
AudioTime	The time segments, given as millisecond intervals, where audio manipulation occurs (start at 0 ms), e.g., $[[300, 1200]]$
AudioSample	The sample segments, given as integer intervals, where audio manipulation occurs, e.g., $[[10, 40]]$
FrameTimeAdjustment	The number of milliseconds to add to the video time segments to adjust time alignment with frames, e.g., 3

3.3.2 REFERENCE MASK

A reference mask is an image used to represent which regions of an image have been manipulated.

For the image localization tasks, the mask is a JPEG 2000 image. A white pixel indicates that the region is not manipulated while a black pixel indicates that the region is manipulated in some way. Each bit plane indicates a

separate manipulation. The reference mask can be filtered according to the types of manipulation a system detects.

3.4   DIRECTORY STRUCTURE

The data directory provided to the performer is organized as follows. In this example (shown below), files from both OpenMFC 2020-2021 (e.g., these filenames start with OpenMFC20) and OpenMFC 2022 (e.g., these filenames start with OpenMFC22) are given.

```
<BaseDir>
        README.txt
        /probe
                {ImageFileName1}.jpg
                {ImageFileName2}.tif
                …
                {VideoFileName1}.avi
                {VideoFileName2}.gif
                …
        /indexes
                OpenMFC20_Image-MD-index.csv
                OpenMFC20_Video-MD-index.csv
                OpenMFC22_ImageSplice-MD-index.csv
                OpenMFC20_Image-DD-index.csv
                OpenMFC20_Video-DD-index.csv
                OpenMFC22_Video-DD-index.csv
                OpenMFC22_Video-StegD-index.csv
        /references
                /IMD
                        OpenMFC20_Image-MD-ref.csv
                        OpenMFC20_Image-MD-ref-journalmask.csv
                        OpenMFC20_Image-MD-ref-probejournaljoin.csv
                        OpenMFC22_ImageSplice-MD-ref.csv
                        /mask
                                {ImageFileName1}.jp2
                                {ImageFileName2}.jp2
                                …
                /VMD
                        OpenMFC20_Video-MD-ref.csv
                        OpenMFC20_Video-MD-ref-journalmask.csv
                        OpenMFC20_Video-MD-ref-probejournaljoin.csv
                /IDD
                        OpenMFC20_Image-DD-ref.csv
                        OpenMFC20_Image-DD-ref-journalmask.csv
                        OpenMFC20_Image-DD-ref-probejournaljoin.csv
                        /mask
                                {ImageFileName1}.jp2
                                {ImageFileName2}.jp2
                                …
```

/VDD

    OpenMFC20_Video-DD-ref.csv
    OpenMFC22_Video-DD-ref.csv

/StegD

    OpenMFC22_Image-StegD-ref.csv

4   SYSTEM INPUT

For a given task, a system's input is the task index file, called <DataSet_Name>-<TaskID>-index.csv and found in the 'indexes' subdirectory.

Given an index file, each row specifies a test trial. Taking the corresponding image(s) or video(s) from the 'probe' directories as input(s), a system performs detection.

4.1   INDEX FILE

The index files contain task ID, file ID of the probe, probe filename, probe width, probe height, device ID, and sensor ID of the specified manipulation detection trial. The probe width and probe height fields are the rendered width and height; for images, this considers the EXIF (exchangeable image file format) rotation field. The device ID is the high provenance capturing device (e.g., camera) of the probe; the sensor ID is a concatenation of device ID and whether the sensor used on the device was the primary or secondary capturing sensor. This is shown in the following example:

TaskID	e.g., "IMD", "VMD"
ProbeFileID	e.g., 001f9af3165a39c9e42aee922f874326
ProbeFileName	e.g., probe/001f9af3165a39c9e42aee922f874326.jpg, probe/00b85e9c14b8c6d02e57255fc71c84c0.mp4
ProbeWidth	e.g., 4912
ProbeHeight	e.g., 3264
ProbeFileSize	e.g., 6075981.0
HPDeviceID	e.g., MK-NEX5T
HPSensorID	e.g., MK-NEX5T_primary or MK-NEX5T_

5   SYSTEM OUTPUT

In the following section, the types of system outputs are defined. The MediScore package[1] contains a submission checker that validates the submission at both the syntactic and semantic levels.

Participants should check their submission prior to sending them to NIST. NIST will reject any submissions that do not pass validation. Consult the MediScore documentation for validation instructions.

5.1   SYSTEM OUTPUT FILE

The system output file must be a CSV file that includes the confidence score and the filename of the output mask (this can be omitted if no mask is required by the task, e.g., VMD).

The filename for the output file must follow the naming convention: <SubID>/<SubID>.csv, where <SubID> is the submission identifier as described in Appendix A.

---

[1] Available at: https://github.com/usnistgov/MediScore/

The system output CSV file for the image detection and localization tasks (e.g., IMD and IDD) must follow the format below:

Col1:	ProbeFileID	e.g., 001f9af3165a39c9e42aee922f874326
Col2:	ConfidenceScore	e.g., 0.8594
Col3:	OutputProbeMaskFileName	e.g., mask/001f9af3165a39c9e42aee922f874326-mask.png [blank if no localization performed]
Col4:	ProbeStatus	i.e., Processed	NonProcessed	FailedValidation
Col5:	ProbeOptOutPixelValue	[blank for OpenMFC evaluation]

The system output CSV file for the image detection only tasks (e.g., StegD) task must follow the format below:

Col1:	ProbeFileID	e.g., 001f9af3165a39c9e42aee922f874326
Col2:	ConfidenceScore	e.g., 0.4837
Col3:	ProbeStatus	i.e., Processed	NonProcessed	FailedValidation

The system output CSV file for the video detection tasks (e.g., VMD and VDD) task must follow the format below:

Col1:	ProbeFileID	e.g., 00b85e9c14b8c6d02e57255fc71c84c0
Col2:	ConfidenceScore	e.g., 0.4837
Col3:	ProbeStatus	i.e., Processed	NonProcessed	FailedValidation

5.2   CONFIDENCE SCORE

The confidence score is any real number in any range that indicates the strength of the possibility that the probe has been manipulated. The scale of the confidence score is arbitrary but must be orderable and consistent across all testing trials with larger values indicating greater chance that the image or video has been manipulated. Those scores are used to generate the performance curve displaying the range of possible operating characteristics.

5.3   PROBE STATUS

The probe status indicates if a particular probe (image or video) was processed or not. A status of "Processed" indicates that the probe was processed, and a confidence score and mask were rendered. A status of "NonProcessed" indicates that probe was not processed due to a system failure of some kind. A status of "FailedValidation" indicates that probe failed the MediScore Validator tool and will be given a score of 0—see details in the MediScore Validator ReadMe file.

The confidence scores for non-processed trials must be 0.

The process for determining which trials are not to be processed must be documented in the system description (Appendix A-a).

5.4   VALIDATION

The ProbeFileID column in the system output <SubID>/<SubID>.csv must be consistent with the ProbeFileID column in the <BaseDir>/indexes/<DataSet_Name>-<TaskID>-index.csv file. The row order may change, but the two ProbeFileID columns must have a one-to-one correspondence.

The value of the ConfidenceScore column in the <BaseDir>/<SubID>.csv file is any real number in the range $[0, 1]$.

6   METRICS DEFINITION

This section describes the metrics that will be used for the evaluation. Two types of metrics are used in the evaluation: score metrics and mask metrics.

6.1   SCORE METRICS

6.1.1   RECEIVER OPERATING CHARACTERISTIC (ROC)

The receiver operating characteristic (ROC) curve is a score metric. Macmillan and Creelman [2] provide detailed information about ROC curves for detection system evaluation. Here is a brief description of the curve. In what follows,

TP stands for True Positive (those correctly detected as manipulated),
FN stands for False Negative (those incorrectly detected as non-manipulated),
FP stands for False Positive (those incorrectly detected as manipulated), and
TN stands for True Negative (those correctly detected as non-manipulated).

The y-axis is the True Positive Rate (TPR), where $TPR = TP/(TP + FN)$; and is also known as **Sensitivity** [3].

The x-axis is the False Positive Rate (FPR), where $FPR = FP/(TN + FP)$; and is also known as FAR (False Acceptance Rate or False Alarm Rate), as well as $1 - \text{specificity}$. Figure 1 illustrates the ROC curve as the red curve.
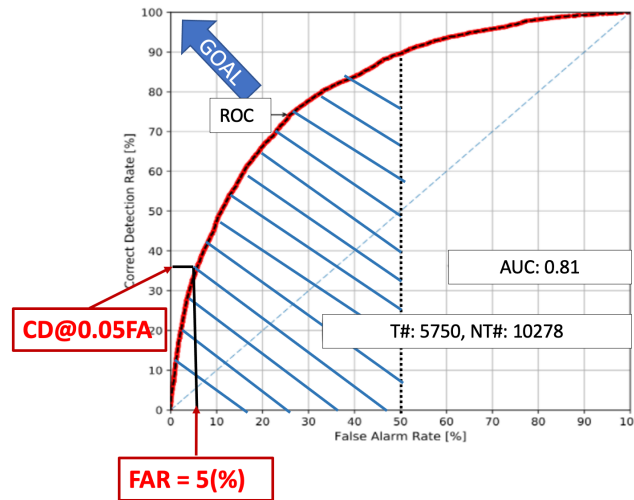

Figure 1: ROC and AUC

6.1.2   AREA UNDER THE ROC CURVE (AUC)

The area under the ROC curve (AUC) is a score metric for the detection system, which is shown as the shaded blue region under the ROC curve in Figure 1. The AUC score quantifies the overall ability of a system to discriminate between two classes. The AUC-value of a system output has a value between 0 and 1.0. A poor-performing system has AUC near 0. A system no better at identifying true positives than random guessing has an AUC of 0.5. A perfect system (no false positives or false negatives) has an AUC of 1.0.

6.1.3 CORRECT DETECTION (CD) AT FALSE ALARM RATE (FAR)

Another score metric used for the detection system is the ROC's Correct Detection (CD) at a specified False Alarm Rate (FAR). It is shown as the red region in Figure 1.

Appendix A   SUBMISSION INSTRUCTIONS

System output and documentation submission to NIST for subsequent scoring must be made on the https://mfc.nist.gov/ web site. The submission protocol, (which is also documented on the https://mfc.nist.gov/#pills-submission-rules) consisting of three steps:
>   (a) preparing a system description and self-validating system outputs,
>   (b) packaging system outputs and system descriptions, and
>   (c) identifying the "system" and transmitting the data to NIST.

The packaging and file naming conventions for OpenMFC rely on **Submission Identifiers** (SubID) to organize and identify the system output files and system description files for each evaluation task/condition. <SubID> is a **user-defined** string that identifies the submission and contains **no spaces or special characters except '_' (underscore) and '-' (hyphen).**

A-a   SYSTEM DESCRIPTIONS

Documenting each system is vital to interpreting evaluation results. As such, each submitted system, determined by unique experiment identifiers, must be accompanied by a system description with the following information.

Section 1.   Submission Identifier(s)

List all the submission IDs for which system outputs were submitted. Submission IDs are described in further detail above.

Section 2.   System Description

A brief technical description of your system.

Section 3.   System Hardware Description and Runtime Computation

Describe the computing hardware setup(s) and report the number of CPU and GPU cores. A hardware setup is the aggregate of all computational components used.

Report salient runtime statistics including: wall clock time to process the index file, wall clock time to index the world data set and the provenance tasks, index size for the world data set, resident memory size of the index, etc.

Section 4.   Training Data and Knowledge Sources

List the resources used for system development and runtime knowledge sources beyond the provided OpenMFC corpora.

Section 5.   References

List pertinent references, if any.

A-b   PACKAGING SUBMISSIONS

Using the SubID, all system output submissions must be formatted according to the following directory structure:

```
<SubID>/
        <SubID>.txt                    The system description file, described in Appendix A-a
        <SubID>.csv                    The system output file, described in Section 5.1
```

/mask		The system output mask directory
	{MaskFileName1}.png	The system output mask file, described in Appendix D-b
	{MaskFileName2}.png	
	…	

As an example, if the earlier team is submitting, their directory would be:

p-baseline_3_imdl/
       p-baseline_3_imd.txt
       p-baseline_3_imd.csv
       /mask

A-c   TRANSMITTING SUBMISSIONS

To prepare your submission, create the previously described directory structure and make a .tar file via the UNIX command 'tar cvf <SubID>.tar <SubID>'. The system output can be uploaded under a new or existing 'System' label. This system label is a longitudinal tracking mechanism that will allow you to track improvements to your specific technology over time. To create new systems, use the 'Add new System' button on the 'Submission Management' page. Once a system is present, click the respective 'Upload' button to initiate a submission. On the 'New submission' form page please specify submission conditions and provide a downloadable file by using the 'Browse' button. Press the 'Submit' button to finalize the submission. The evaluation scoring backend will then queue your submission for download and processing. You can check the status of your submission anytime by going to the 'Submission Management' page.

For detailed information about submission management see the online documentation at
https://mfc.nist.gov/help/submissions.

Please submit your files in time for us to deal with any transmission errors that might occur well before the due date (https://mfc.nist.gov/#pills-schedule) if possible. Note that submissions received after the stated due dates for any reason will be marked late.

The Media Forensics Challenge evaluation infrastructure uses comma-separated values (CSV) formatted files with an initial field header line as the data interchange format for all textual data. The EBNF structure used by the infrastructure is as follows:

```
CSVFILE        :== <HEADER> <DATA>*
<HEADER>       :== <TEXT_STRING> {"|" <TEXT_STRING> }* <NEWLINE>
<DATA>         :== <TEXT_STRING> {"|" <TEXT_STRING> }* <NEWLINE>
```

An example of the CSV content is as follows (a table and shadow are used to align the column for visualization purposes, there is no physical space between columns before the vertical bar):

City	State	Visited	VisitDate	Population	…
Akron	Ohio	N		197633	…

The first data record in the files is a header line. The header lines are required by the evaluation infrastructure and the field names for the index file and the system output file are dictated by specified tasks.

Each header and data record in the table is one line of the text file. Each field value is a column and is separated from the next value with a vertical bar.

The DetectionScorer script calculates the performance measures of AUC (see Section 6.1.2) and equal error rate (EER) based on a system's output (e.g., confidence scores) for the manipulation and splice detection tasks. Two files are produced. The first is a CSV file containing a report table. The report table contains the measures AUC, EER, and the confidence interval for the AUC (AUC_CI). The second output is an Adobe PDF file containing a graphical plot. The plot displays the ROC (see Section 6.1.1) from the results of the algorithm performance as well as the AUC. The AUC can be partial (up to a certain FAR value) or full (when FAR value is set to 1.00).

Here is a test case.

```
python DetectionScorer.py -t manipulation --
refDir ../../data/test_suite/detectionScorerTests/ -r reference/OpenMFC2020-imdl-
ref.csv -x reference/OpenMFC2020-imdl-index.csv --
sysDir ../../data/test_suite/detectionScorerTests/baseline -s
Base_OpenMFC2020_imdl_ImgOnly_p-copymove_01.csv --outRoot ./testcases/OpenMFC20_001 --
ci --display
```

Table 2: Example of Report Table Output for Test Case

AUC	FAR_STOP	EER	AUC_CI_LOWER	AUC_CI_UPPER
0.679533	1	0.328889	0.620826	0.735491

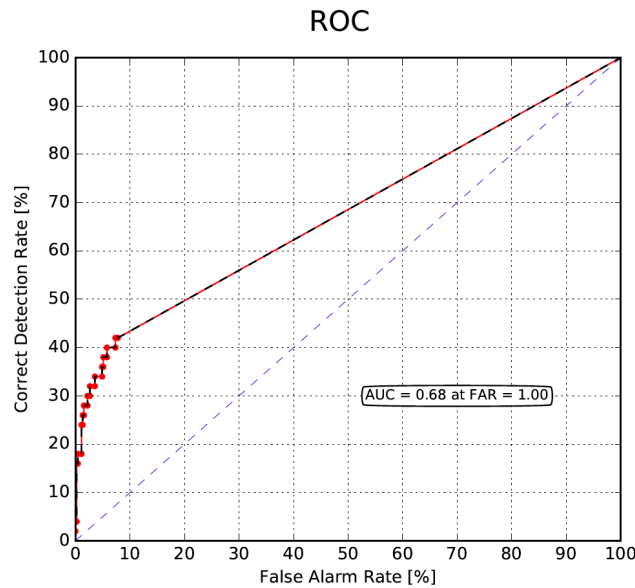**Figure 2: Example of Graphical Output for Test Case**

OpenMFC supports Image Manipulation Localization (IML) and Image Deepfake Localization (IDL) tasks.

D-a   TASK

If the test image is detected to be manipulated, the IML or IDL task will spatially localize the manipulations and provide system output mask image. The system output mask image must be relative to the input test image and indicate the confidence of each location (pixel) was manipulated. If the mask image for a trial is detected by a system to find no localizable content change, the mask can be omitted and is assumed to be empty.

D-b   SYSTEM OUTPUT MASK

The system output mask directory contains the system output of the mask file for the IML or IDL tasks. The directory path and mask filename use the following convention: <SubID>/mask/{MaskFileName}.png for masks, where it is optional to name the mask filenames as {ProbeFileID}-mask.png.

The system should output a mask image to represent the detected region(s) of the manipulation for the IML or IDL tasks. The size of the mask image must be the same size as the probe image. The mask image must be a single channel (grey) image in PNG format. Color images and images with an alpha channel will not be evaluated. For each pixel location in the input image, the system must use a one-byte integer number between 0 (black) and 255 (white) to indicate whether that pixel has been manipulated: smaller numbers indicate a greater chance that the pixel in this location has been manipulated and larger numbers indicate a lesser chance of manipulation. Both binary and grey-scale masks can be evaluated. For binary masks, the system output image's pixels only have two values: 0 (manipulated) and 255 (not manipulated). For grey-scale masks, the mask scorer will report the optimum MCC over all thresholds.

D-c   VALIDATION RULES FOR MASK FILES

Each MaskFileName in the system output file, <SubID>/<SubID>.csv, must exist in the '<SubID>/mask' directory and be readable as a PNG file for image masks. Each MaskFileName in the system output file, <SubID>/<SubID>.csv, must have the same size as its corresponding original image defined in the system output file.

D-d   REFERENCE MASK

The reference mask for each true manipulation with localized content change is a layered reference mask[2], a lossless JPEG 2000[3] image in which, on each layer, a white pixel indicates the region has not been manipulated and a black pixel indicates the region has been manipulated. The reference mask for each true manipulation without localized content change is a reference mask, a JPEG 2000 image in which each pixel is white; that is, if there is no localized content change in a true manipulation, the mask is completely white. Each bit plane indicates a separate manipulation. Not all manipulations require localization output. Global operations affecting the entire image are not required for localization output because then the entire image is marked as manipulated; for example, a clone operation does require localization output while global histogram normalization does not.

---

[2] Defined in Section 3.3.2.
[3] https://jpeg.org/jpeg2000/index.html

Figure 3 shows an example of an image manipulation detection and localization trial. In this trial, image (a) is the original image. Image (b) is created by cloning the umbrella and splicing a polar bear into the image. Each manipulation in the trial is indicated by a different color in the reference mask as shown in image (c). The cloning of the umbrella is indicated by purple, and the splicing in of the polar bear is indicated by pink. The image (b) is the probe used in the test evaluation.

(a) The base (original) image   (b) The probe (manipulated) image   (c) The mask image

Figure 3: An example of a trial for the image manipulation detection task

D-e   MASK METRICS

The metric for measuring image manipulation localization performance is the Optimum Matthews Correlation Coefficient (MCC) as defined in Section D-e-ii.

Masks are only evaluated on trials in which the specified manipulation occurred. If the system output mask for a trial was not deemed worthwhile and was therefore omitted, a mask score of -1 will be given for that trial.

D-e-i   DEFINITION OF REGIONS

Figure 4 is a visualization of the different mask regions used for mask image evaluations. Figure 4-a shows the reference mask; Figure 3-b shows system dilation and erosion operations; Figure 3-c shows system weights; Figure 4-d shows the system output mask. Figure 4-e shows the resulting mask regions.

Because of the complexity of the problem, a region around the mask will not be scored. To create this no-score region, dilation and erosion operations will be performed on the reference mask. Figure 4-b illustrates the dilation and erosion operations on the reference mask from Figure 4-a. Figure 4-c illustrates the different regions of the reference mask after the dilation and erosion operations from Figure 4-b. The solid black area in the middle, the remainder after the erosion operation, is denoted as the GT region, i.e., $GT = Erosion(M_r)$ where M_r is the black region in Figure 4-a. This is the region that will be scored as the correct manipulation region. The solid white region, the remainder after the dilation operation, is denoted as the $NotGT$ region, i.e., $NotGT = M_r - Dilation(M_r)$. This is the region that will be scored as the correct non-manipulated region. The shaded purple region between the GT and $NotGT$ regions, the result of the dilation and erosion operations, is the $NoScore$ region, i.e., $NoScore = Dilation(M_r) - Erosion(M_r)$. Any pixels in this region will be ignored for scoring purposes.

When evaluating the system output mask, Figure 4-d, using the reference mask (post dilation and erosion), Figure 4-e, the pixels are classified into the following regions based on the concepts described in [4]. Refer to Figure 4-e for all the classified regions. For spatial video masks, these regions will be taken across all frames in an interval of manipulated frames.

- True Positive (TP, also called Correct Detection, CD): The reference mask indicates it is manipulated, and the system also detected it as manipulated. The region is shown in solid green.
- False Negative (FN, also called Missed Detection, MD): The reference mask indicates it is manipulated, but the system did not detect it as manipulated. The region is shown in solid red.

- False Positive (FP, also called False Alarm, FA): The reference mask indicates it is not manipulated, but the system detected it as manipulated. The region is shown in solid orange.
- True Negative (TN, also called Correct Rejection, CR): The reference mask indicates it is not manipulated, and the system also does not detect it as manipulated. The region is shown in solid white.
- No-Score (NS): The region of the reference mask not scored, the result of the dilation and erosion operations. The region is shown in cross-hatched purple.
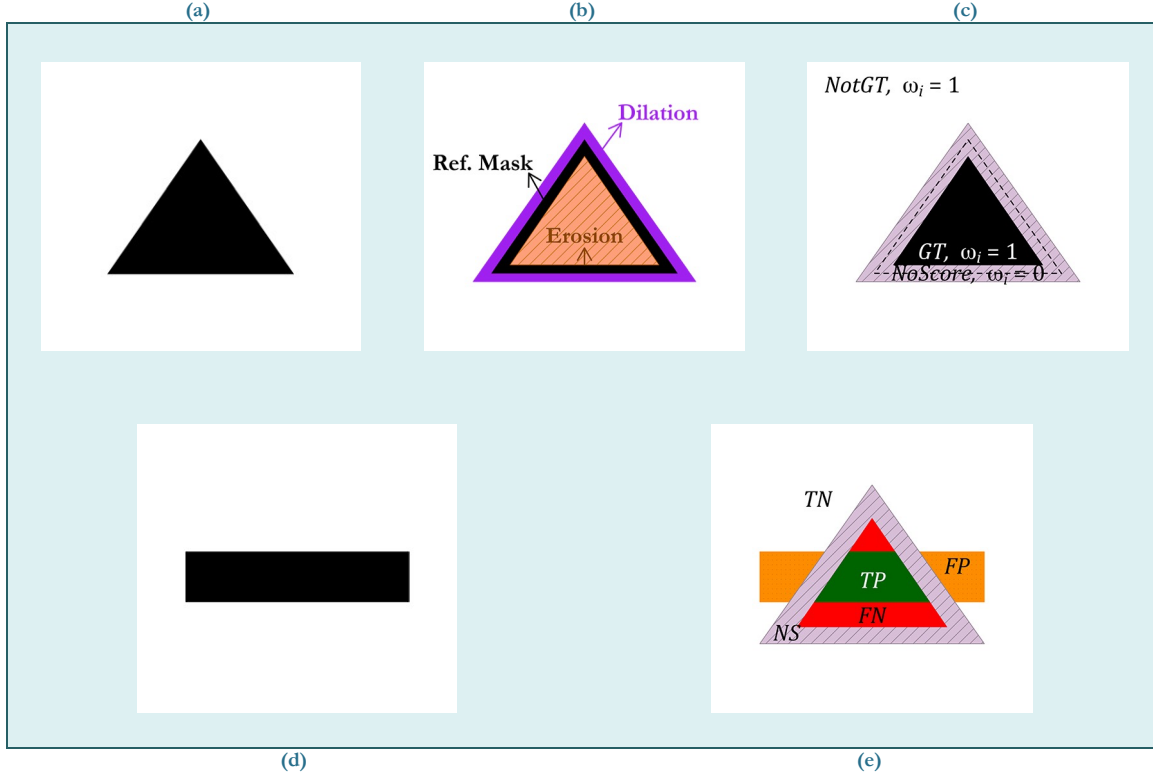

Figure 4: Mask Regions

The mask metrics used is the Matthews Correlation Coefficient (MCC) [4]. Refer to Figure 4 and the region descriptions above for the definitions of TN, TP, FN, and FP.

$$\text{MCC} = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

MCC ranges from -1 to 1. If MCC $= 1$, there is perfect correlation between the reference and system output masks. If MCC $= 0$, there is no correlation between the reference and system output masks. If MCC $= -1$, there is perfect anti-correlation between the reference and system output masks. If the denominator is zero, then MCC is set to 0.

Implicit in the several mask metrics is the specification of a threshold value for which the system determines a given pixel to be modified. Systems are expected to provide a single threshold to be used for all masks; however, other thresholds using the reference data as an oracle can be used. The following names designate the

rule used to determine the threshold, which separates manipulated and non-manipulated pixels. These names will be used as a prefix to the measurement name, for example "Actual MCC".

- Actual – The metric is calculated using the system-provided global threshold
- Maximum (Minimum) – The metric is calculated using a single ideal global threshold found by computing metric scores over all thresholds over all masks
- Optimum – The metric is calculated using an ideal mask-specific threshold found by computing metric scores over all thresholds

D-e-iv   MASK SCORING EVALUATION CONDITION

As stated earlier, if performing localization, only the masks of known manipulated images will be evaluated. If no mask image is given for a trial of a known manipulated image, an MCC score of -1 will be assigned for that trial. An example is illustrated in Table 3 below.

Table 3: An Example of Outcome of Scoring System Output Masks

Image File Name	Manipulated?	Confidence Score	Mask File Exists?	MCC Score
001f9af3165a39c9e42aee922f874326.jpg	N	0.3126	N	N/A
005d09c11ef1c13307e2519090695adf.png	N	0.7305	Y	N/A
0073243b5279a64dbbbc10278a25b4ce.png	N	0.2546	N	N/A
007ea872e68b3de3ed960e8a0762dbd2.tif	N	0.3939	N	N/A
009c4442e337b90a52f7aad6261b4918.bmp	N	0.8453	Y	N/A
00dc25da70dc8c7b42dd785bebba8ad1.nef	Y	0.7603	Y	0.591
00f5540e736c5f5c5dd598bd7617a02d.png	Y	0.7350	Y	0.864
01004acc7d89272c05b19b31861071d4.jpg	Y	0.1707	N	-1
0101f0966fb444dbdf9afc96149f6382.jpg	Y	0.2307	N	-1
010fdced5feaa64d42178a8c664eacce.tif	Y	0.6041	Y	0.394

7   BIBLIOGRAPHY

[1] J. Newman, L. Lin, W. Chen, S. Reinders, Y. Wang, M. Wu and Y. Guan, "Stegoappdb: a steganography apps forensics image database," *Electronic Imaging,* no. 5, pp. 536-1, 2019.

[2] N. A. Macmillan and C. D. Creelman, Detection Theory: A User's Guide, Psychology Press, 2004.

[3] J. Yerushalmy, "Statistical problems in assessing methods of medical diagnosis with special reference to x-ray techniques," *Public Health Reports,* vol. 62, no. 2, p. 1432–39.

[4] B. W. Matthews, "Comparison of the predicted and observed secondary structure of T4 phage lysozyme," *Biochimica et Biophysica Acta (BBA) - Protein Structure,* vol. 405, no. 2, pp. 442-451, 1975.