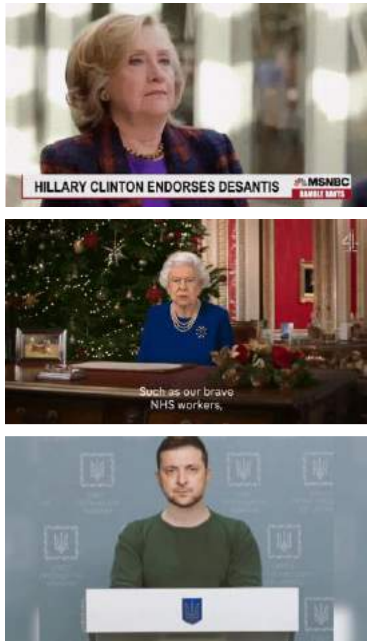
VideoFACT: Detecting Video Forgeries using Attention, Scene Context, and Forensic Traces

Tai D. Nguyen, Shengbang Fang, Matthew C. Stamm
Multimedia & Information Security Lab
Drexel University, Philadelphia, USA

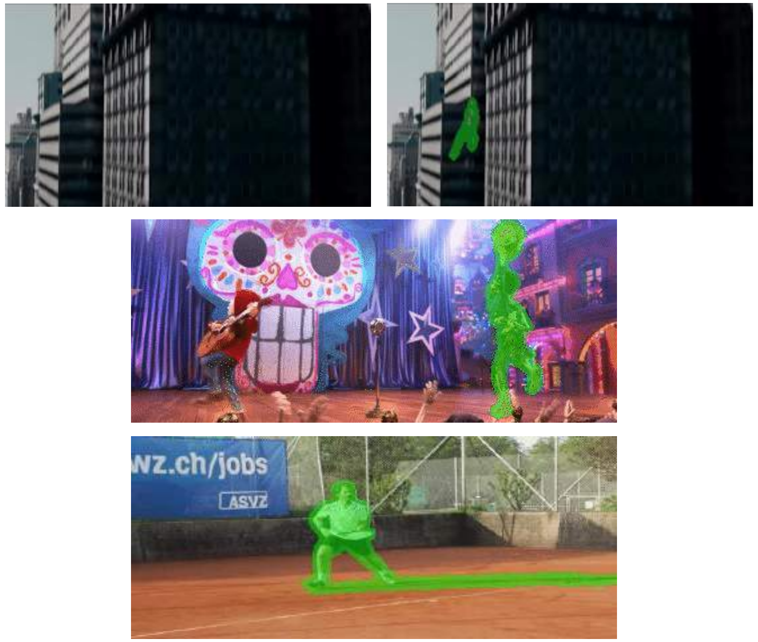
Introduction

- Misinformation is a threat to society
- Many ways to create fake videos

Deepfakes



Inpainting



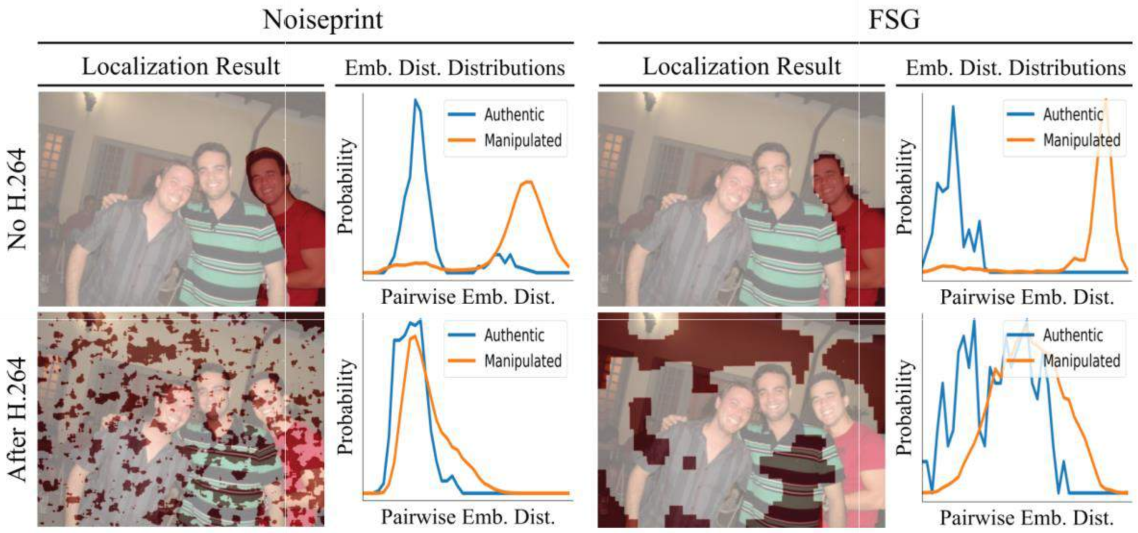
Splicing/Editing

Problem

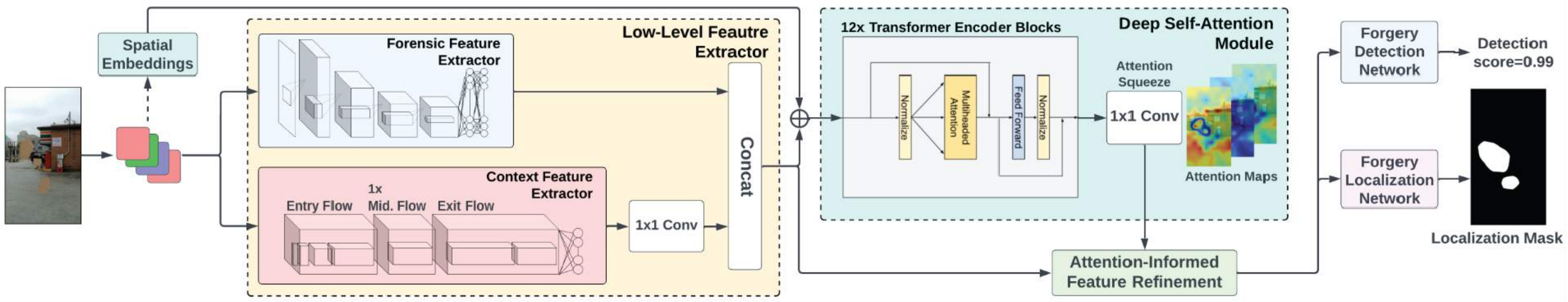
- Existing media forensic techniques fall into two categories:
  - “Specialist” detectors for video (deepfake detectors, inpainting detectors)
  - “Generalist” detectors for images
- Problem: No generalist detector for video, capable of identifying many forgery types
  - Existing detectors for video only work on one manipulation type
  - Image manipulation detectors all fail on video!

Effects of Modern Video Coding

- Modern Video Codecs (H.264) adversely affects anomaly-based forgery detectors
  - These detectors search for inconsistencies in forensic traces
  - H.264 encodes each macroblocks differently
    - Naturally induces inconsistencies
    - Causing all image detectors to fail!

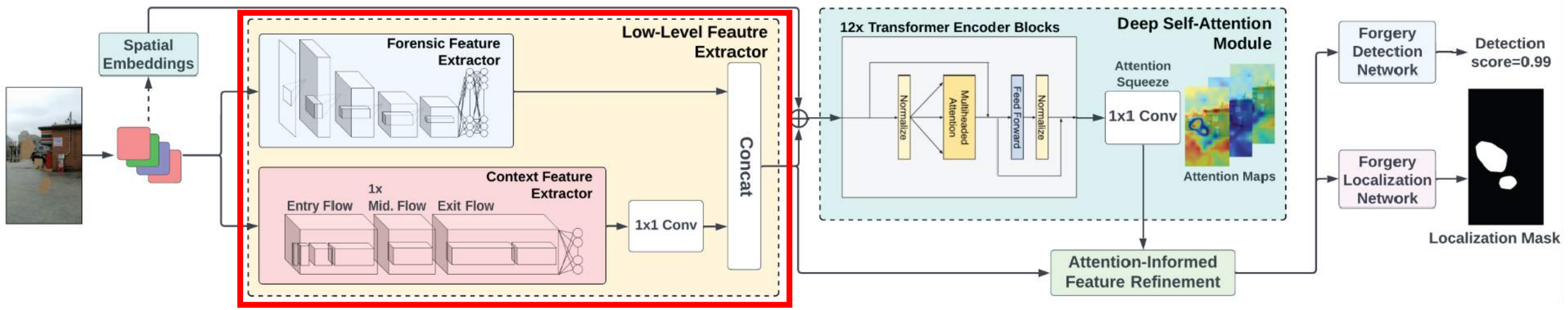


Proposed Approach



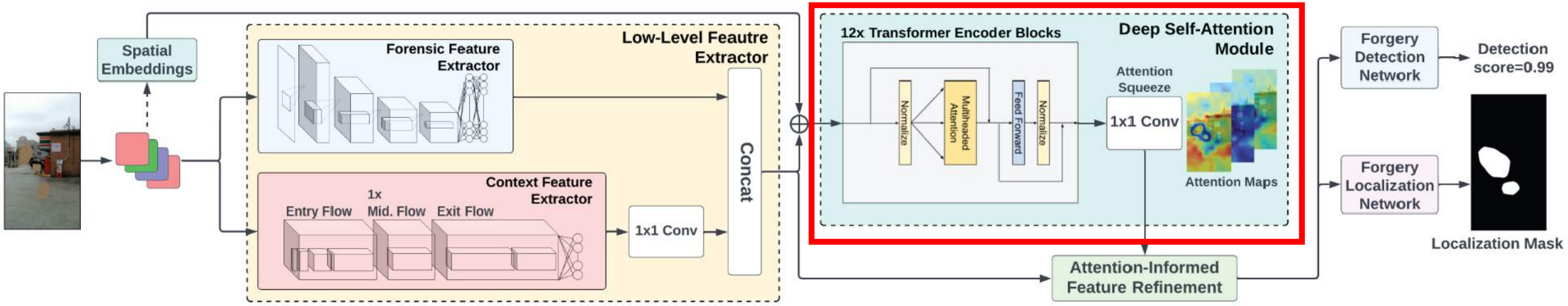
- We propose VideoFACT, a network capable of detecting and localizing a wide variety of video forgeries
  - Capture forensic features specific to video
  - Learn new context features to control for variation in forensic traces caused by video coding
  - Create a set of joint feature embeddings that are analyzed using a deep self-attention module
  - Refine the joint feature embeddings using attention maps
  - Produce accurate decisions using separate subnetworks for detection and localization

Low-Level Feature Extraction



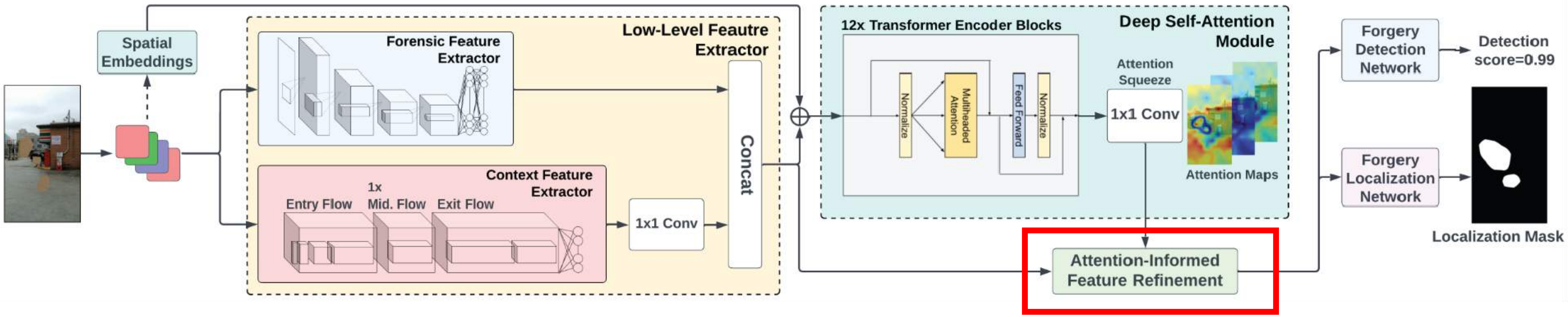
- We use two types of feature embeddings
  - Forensic Feature Embeddings designed to capture traces specific to video
  - Context Feature Embeddings learned to control for variation in forensic traces caused by video coding
  - Concatenate to make Joint Feature Embeddings

Deep Self-Attention Module (DSAM)



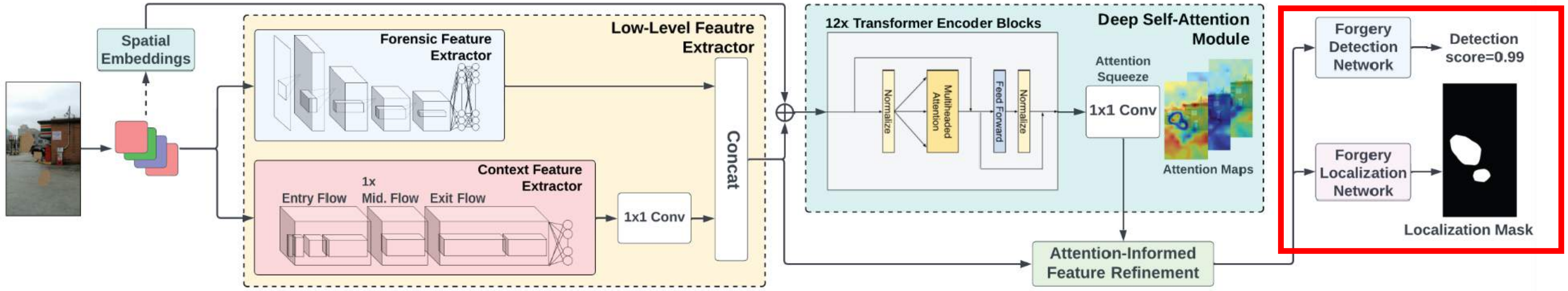
- This module's purpose is to
  - Estimate the quality and relative importance of local embeddings by
    - De-emphasizing regions of low-quality traces
    - Emphasizing regions with high-quality traces and potential manipulation

Attention-Informed Feature Refinement



- This module refines the Joint Feature Embeddings by
  - Using attention maps from DSAM to scale embedding vectors by its respective weights
  - Enabling accurate decision making
  - Minimizing false alarms due to video coding

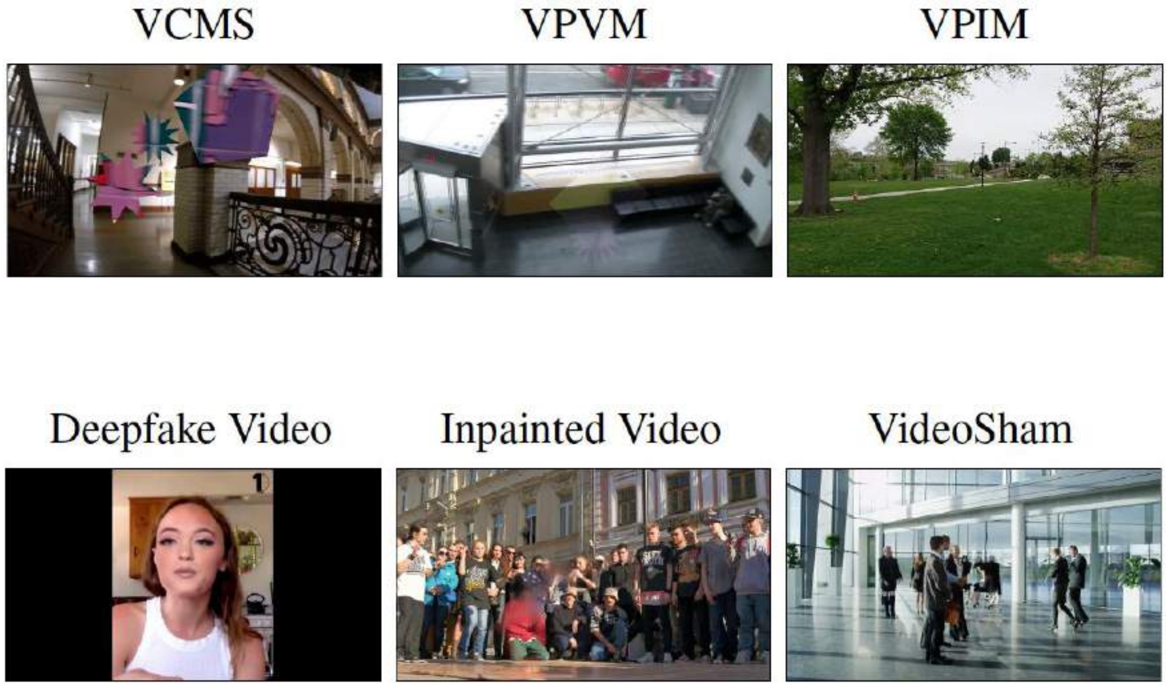
Detection and Localization



- The Detection Subnetwork
  - Combines refined embeddings into detection score
- The Localization Subnetwork
  - Distills refined embeddings into patch-level localization map
  - Converts patch-level map into pixel-level map using bilinear interp. & smart thresholding

Datasets

- No public general video forgery datasets
  - Only Adobe VideoSham (WACV 2023) for evaluation
  - None for training
- "Standard Manipulation" datasets created by us
- "In-the-Wild" datasets
  - AI-Based Inpainting
    - Created by us using E2FGVI & FuseFormer algorithms
  - Deepfakes
    - DeepFaceLab deepfakes created by us
    - FaceForensics++, Deepfake Detection Dataset (DFD)



Multi-Stage Training Protocol

- We employ a multi-stage training protocol
  - Improve generalization to unseen manipulations
  - Improve convergence during training
  - Protocol:
    - Pretrain FFE on Camera Identification task using VideoASID dataset
    - Progressively train on increasingly challenging manipulations (VCMS -> VPVM -> VPIM)
    - *Not* training on deepfake or inpainting

Stage	Dataset	Opti-mizer	Epochs	Initial Lr	Decay rate	Decay step
1	A	SGD	6	1.0e−4	0.75	2
2	B	SGD	6	8.5e−5	0.85	2
3	C	SGD	23	8.5e−5	0.85	2
4	A, B, C	SGD	10	8.5e−5	0.85	2
5	A, B, C, D, E, F	SGD	9	5.0e−5	0.85	2

Table 1. Training parameters for different training stages of our model. We denote: A=VCMS, B=VPVM, C=VPIM, D=ICMS, E=IPVM, F=IPIM.

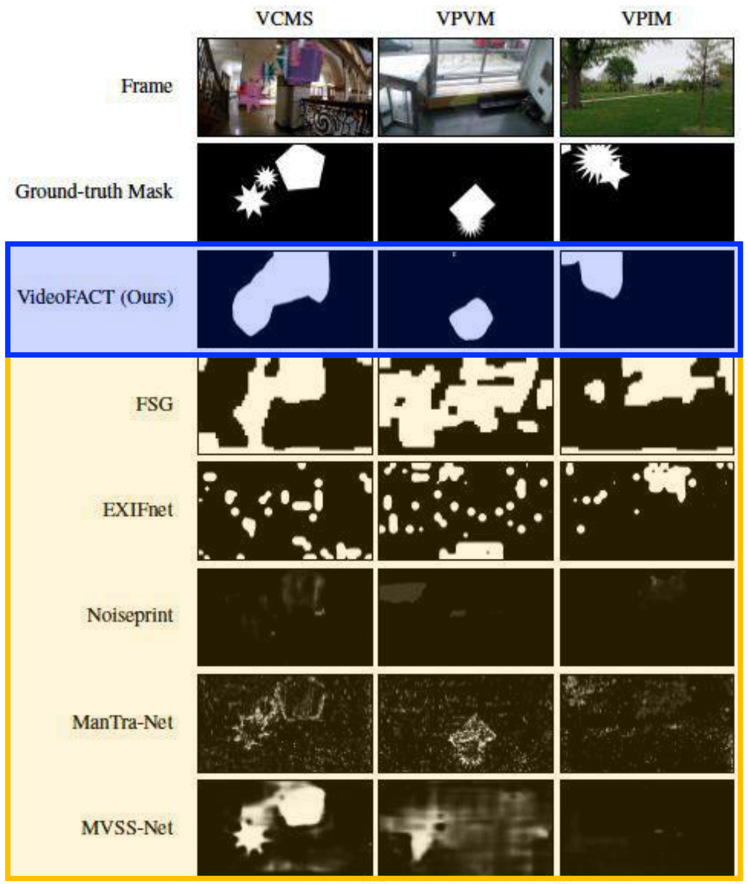
Experiments

- We benchmarked against
  - SOTA general image forgery detectors
  - Specialized video manipulation detectors
- We benchmarked on
  - Set A: Standard Video Manipulations Datasets
  - Set B: In-the-Wild Video Manipulation Datasets
- We reported performance with
  - mAP and accuracy for detection
  - MCC and F1 for localization

Results – Splicing & Editing

- Very strong detection & localization performance
  - VCMS – Splicing
  - VPVM – Editing
  - VPIM – Editing (Invisible)
- Existing detectors largely fail

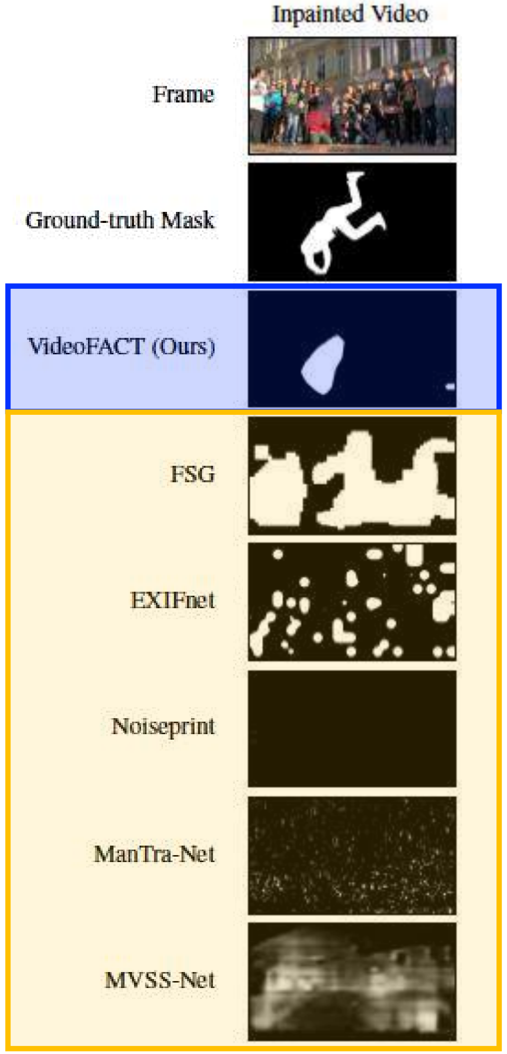
Method	VCMS				VPVM				VPIM			
	Det. mAP	*Det. ACC*	*Loc. MCC*	*Loc. F1*	*Det. mAP*	*Det. ACC*	*Loc. MCC*	*Loc. F1*	*Det. mAP*	*Det. ACC*	*Loc. MCC*	*Loc. F1*
FSG [40]	0.445	0.497	0.001	0.064	0.431	0.480	0.004	0.067	0.485	0.494	0.011	0.065
EXIFnet [26]	0.610	0.502	0.208	0.230	0.568	0.501	0.213	0.236	0.509	0.500	0.026	0.124
Noiseprint [12]	0.521	0.500	0.041	0.030	0.495	0.500	0.012	0.013	0.511	0.500	0.010	0.010
ManTra-Net [58]	0.451	0.500	0.079	0.114	0.526	0.500	0.110	0.145	0.513	0.500	0.025	0.064
MVSS-Net [8]	0.883	0.602	0.545	0.557	0.644	0.529	0.267	0.279	0.482	0.492	0.018	0.042
VideoFACT	0.995	0.987	0.530	0.526	0.980	0.950	0.676	0.697	0.869	0.797	0.515	0.547



Results - Inpainting

- Baseline VideoFACT: not trained on any inpainting data
  - Good detection & localization results
- VideoFACT-FT: fine tuned using very small number of examples
  - Excellent detection & localization results
- Existing approaches largely fail

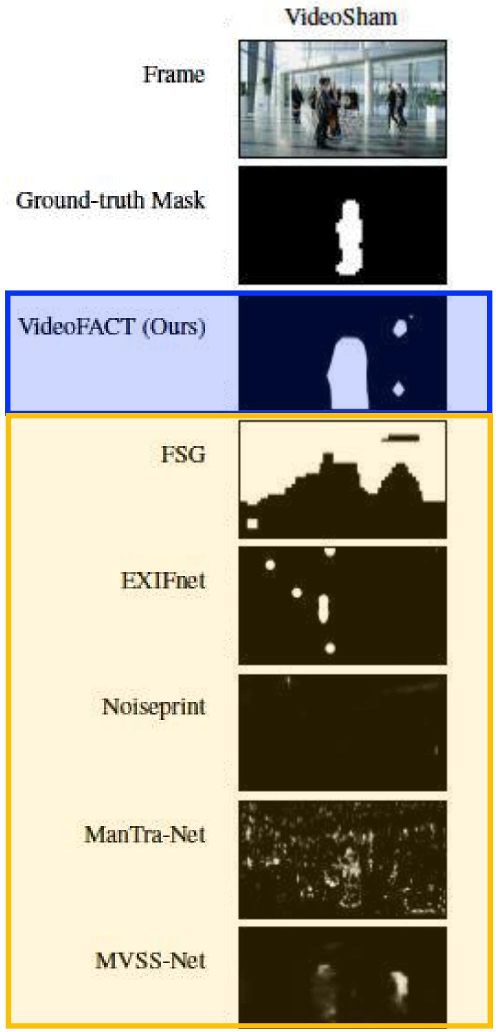
Method	E2FGVI Inpainted Videos				FuseFormer Inpainted Videos			
	Det. mAP	*Det. ACC*	*Loc. MCC*	*Loc. F1*	*Det. mAP*	*Det. ACC*	*Loc. MCC*	*Loc. F1*
FSG [40]	0.386	0.452	0.208	0.302	0.351	0.484	**0.241**	**0.290**
EXIFnet [26]	0.635	0.501	0.160	0.244	0.506	0.507	0.146	0.225
Noiseprint [12]	0.601	0.500	0.091	0.232	0.471	0.500	0.001	0.200
ManTra-Net [58]	0.499	0.500	0.009	0.055	0.613	0.500	0.031	0.204
MVSS-Net [8]	0.341	0.435	0.058	0.227	0.230	0.359	0.029	0.206
VideoFACT	**0.782**	**0.687**	**0.225**	**0.309**	**0.652**	**0.527**	0.118	0.237
VideoFACT-FT	**0.908**	**0.820**	**0.411**	**0.445**	**0.948**	**0.846**	**0.361**	**0.411**



Results – Adobe VideoSham

- VideoSHAM contains multiple types of manipulation
  - Color swapping, object add/remove, text add/remove, etc.
- VideoFACT not trained or finetuned on any of this data
  - Strongest reported results
- Existing approaches largely fail

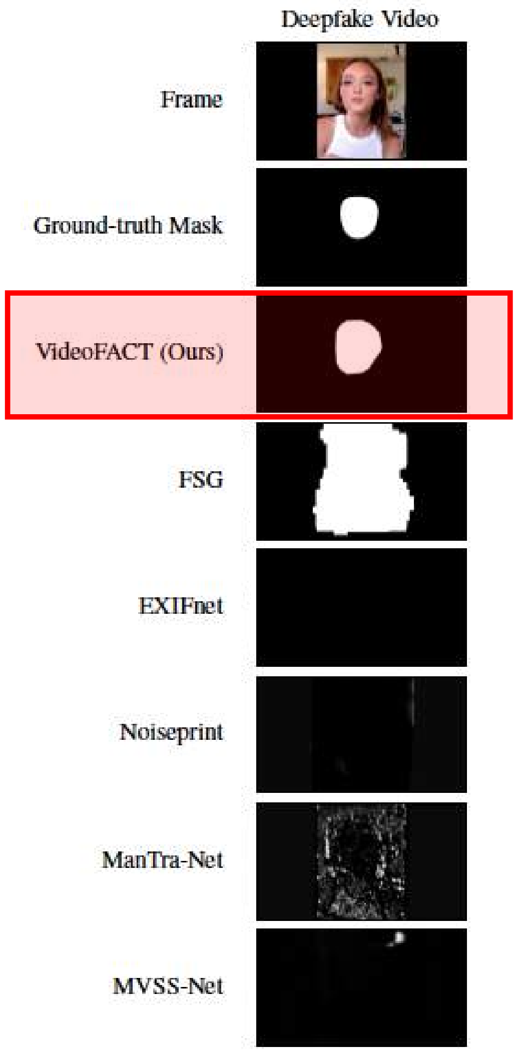
Method	VideoSham [42]			
	Det. mAP	*Det. ACC*	*Loc. MCC*	*Loc. F1*
FSG [40]	0.596	0.538	0.162	0.246
EXIFnet [26]	0.584	0.555	0.148	0.246
Noiseprint [12]	0.422	0.447	0.034	0.206
ManTra-Net [58]	0.551	0.553	0.009	0.058
MVSS-Net [8]	0.595	0.449	0.142	0.096
VideoFACT	**0.691**	**0.656**	**0.193**	**0.312**



Results - Deepfakes

- Baseline VideoFACT performance is mixed
- VideoFACT-FT: fine tuned 10% of DFD & FF++ dedicated training data
  - Excellent detection & localization results
- VideoFACT-FT outperforms existing approaches
  - Splicing detectors largely fail
  - Outperforms existing deepfake detectors on this experiment

Method	DeepFaceLab Deepfake Videos				DFD [14]				FF++ [49]			
	Det. mAP	*Det. ACC*	*Loc. MCC*	*Loc. F1*	*Det. mAP*	*Det. ACC*	*Loc. MCC*	*Loc. F1*	*Det. mAP*	*Det. ACC*	*Loc. MCC*	*Loc. F1*
FSG [40]	0.450	0.515	0.204	0.137	0.449	0.325	0.097	0.043	0.509	0.519	0.144	0.113
EXIFnet [26]	0.447	0.492	0.180	0.133	0.489	0.258	0.095	0.051	0.487	0.519	0.141	0.073
Noiseprint [12]	0.591	0.500	0.010	0.062	0.489	0.252	0.000	0.021	0.486	0.518	0.000	0.066
ManTra-Net [58]	0.450	0.500	0.004	0.042	0.476	0.253	0.017	0.025	0.504	0.514	0.070	0.091
MVSS-Net [8]	0.464	0.498	0.199	0.189	**0.513**	**0.532**	**0.152**	**0.108**	0.499	0.487	0.133	0.164
VideoFACT	**0.666**	**0.648**	**0.415**	**0.410**	0.468	0.444	0.081	0.077	**0.529**	**0.519**	**0.160**	**0.167**
VideoFACT-FT	**0.988**	**0.922**	**0.745**	**0.732**	**0.937**	**0.804**	**0.536**	**0.490**	**0.916**	**0.837**	**0.661**	**0.645**
E.ViT [10]	0.896	0.805	N/A	N/A	0.811	0.737	N/A	N/A	0.764	0.676	N/A	N/A
CCE.ViT [10]	0.962	0.837	N/A	N/A	0.816	0.761	N/A	N/A	0.796	0.719	N/A	N/A
CNN Ensemble [6]	0.936	0.857	N/A	N/A	0.829	0.745	N/A	N/A	0.713	0.672	N/A	N/A



Results – Key Takeaways

- Baseline Performance
  - Able to detect & localize a wide variety of forgery types (splicing, editing, deepfakes, inpainting)
  - Significantly outperforms SOTA image detectors
- Targeted Manipulation Performance
  - Can massively improve performance by fine-tuning on small amount of data of a specific manipulation
  - Significant performance gain for inpainting & deepfake detection
  - Achieve equivalent or better performance than SOTA specialists

**Important because manipulation is not known a priori in real world*

Working Modes

- Our network performs best when
  - The falsified region is larger than our analysis window size (128 x 128 pixels)
  - The forgery and its surrounding do not suffer from poor lighting/texture
  - Color swapping in objects can be challenging
  - Network’s input size is limited to 1080p video resolution

Ablation Study

Setup	Component					VideoSham			
	FFE	*CFE*	*Trans-former*	*Attn. maps*	*Data comb.*	*Det. ACC*	*Det. mAP*	*Loc. F1*	Loc. MCC
Proposed	+	+	+	3	Add	**0.656**	**0.691**	**0.258**	**0.168**
No FFE	—	+	+	3	Add	0.610	0.646	0.209	0.118
No CFE	+	—	+	3	Add	0.586	0.635	0.163	0.043
No DSAM	+	+	—	—	—	0.601	0.626	0.144	0.000
No Transformer	+	+	—	3	Add	0.533	0.538	0.140	0.048
No Attention Squeeze	+	+	+	—	—	0.622	0.656	0.254	0.120
1 Attention Map	+	+	+	1	Add	0.610	0.655	0.175	0.121
10 Attention Maps	+	+	+	10	Add	0.622	0.676	0.212	0.127
Diff. Feat. Refine	+	+	+	3	Concat	0.614	0.684	0.162	0.091

Table 4. Ablation study of the components in our proposed network and their performance evaluations.

Thank you!

Drexel
UNIVERSITY

misl
multimedia &
information
security
lab