# What makes fake images detectable? Understanding properties that generalize

Lucy Chai, David Bau, Ser-Nam Lim, Phillip Isola

European Conference on Computer Vision (ECCV), 2020

Presented by Bhanuka Mahanama

October 17, 2022

# Which image is fake?

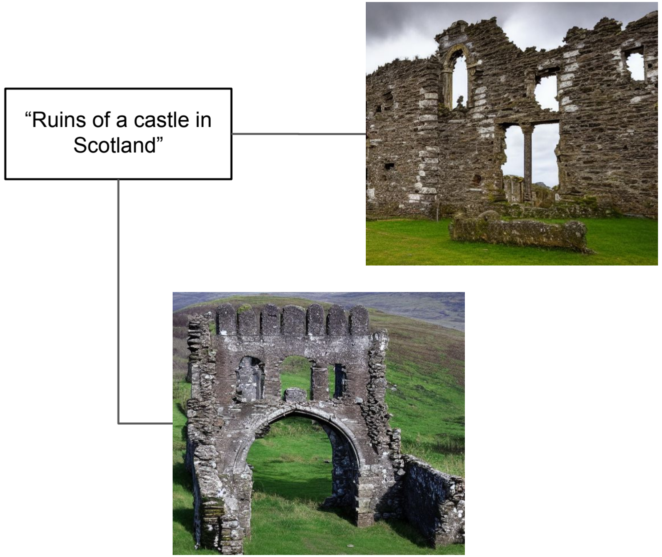
Library of Congress

BBC

# Types of Fake Images

**Spliced images:** Combine multiple images to form the composite image


**Synthesized images:** Generate images using random noise/text



# How to detect fake images?

- Compare against similar images
- Inspect for signs for manipulation
  - Image content
  - Metadata
  - Timestamps
- Domain knowledge

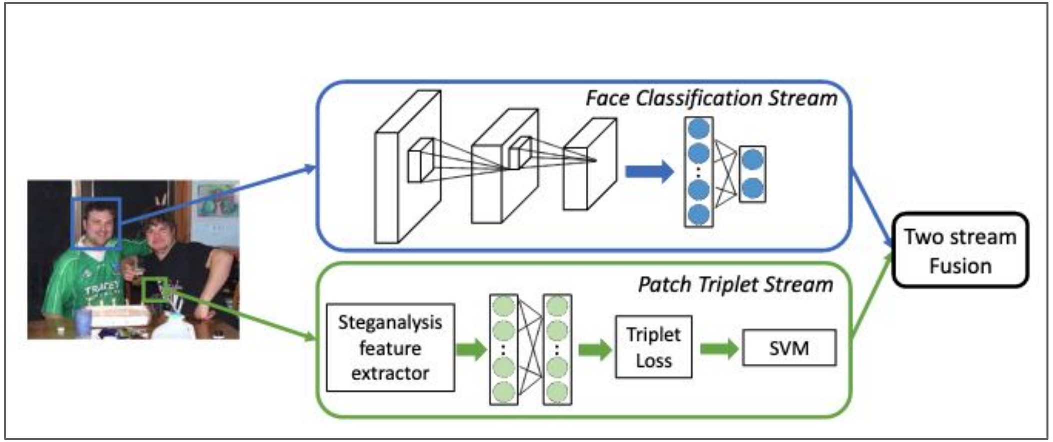
Problems:

- Time consuming
- Does not scale
  - Easy to generate fake images

Library of Congress

# Automated Image Classification

- Consistency throughout image
  - Metadata
  - Low level artifacts (traces of resampling)
  - Similar embeddings
  - Using image features
- Deep learning based
  - RGB images
  - Alternative image representations



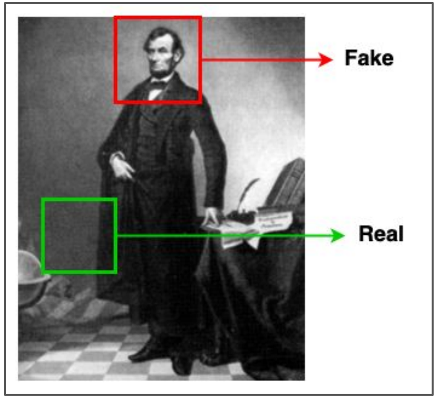

Similar embeddings (Zhou et al., 2017)

# What's the catch

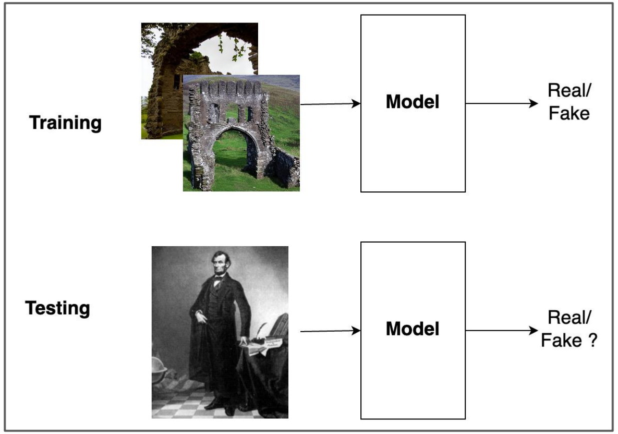
It's straightforward to train a fake/not-fake classifier

But challenging

- Generalize
    - Ability to classify unseen data
- Localize manipulations
    - Identify manipulated regions




Localizing manipulations



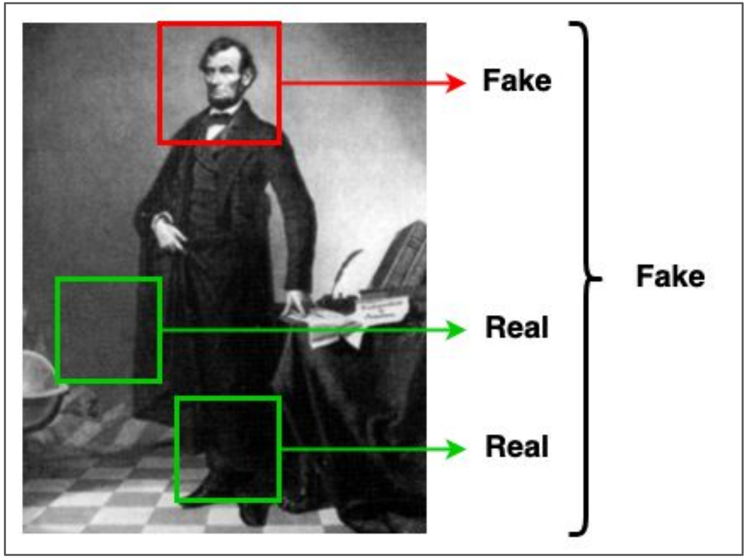

Classifier generalizability

# Solution = Patch Forensics

- Use patches of the image
  - Classify each patch fake/fake-not
  - Ensemble and classify the whole image
- Generalization:
  - No global features
  - Identify local manipulated regions
- Localization:
  - Fake patch = manipulated region

Additional benefits

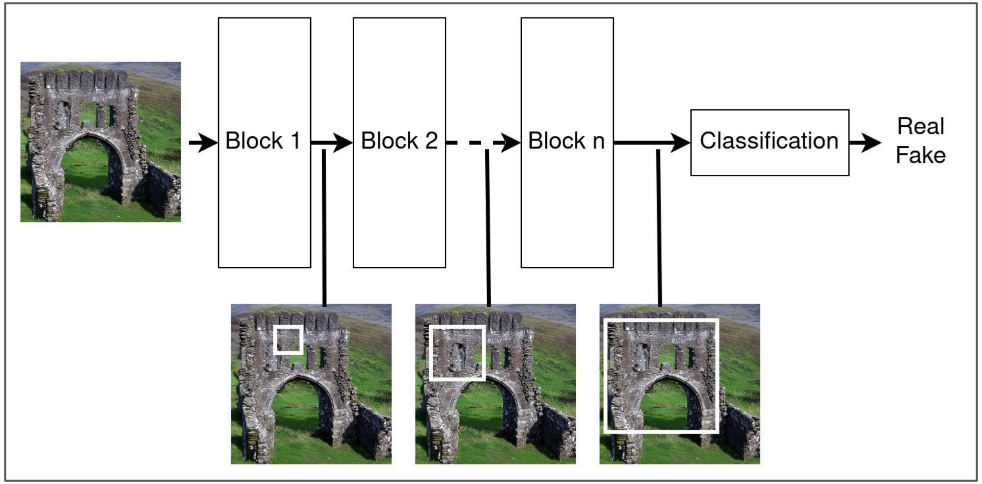
- Shallower models
- Explainability
  - Identify the manipulated regions




Patch forensics: Classification via patches

# Model Architecture

- Deep learning models
  - Series of modules
  - Progressively extract features
  - Receptive field => feature region
- Truncate early
  - Smaller receptive field
  - Local features
- Truncate later
  - Larger receptive field
  - Global features
- Truncate at early layers
- Use output to predict fake/true




Receptive fields with depth of a neural network

# Dataset Processing

Challenge: Minimizing the effects of image processing

Solution:

- Apply same transformations for real images
- Save all images using identical pipelines

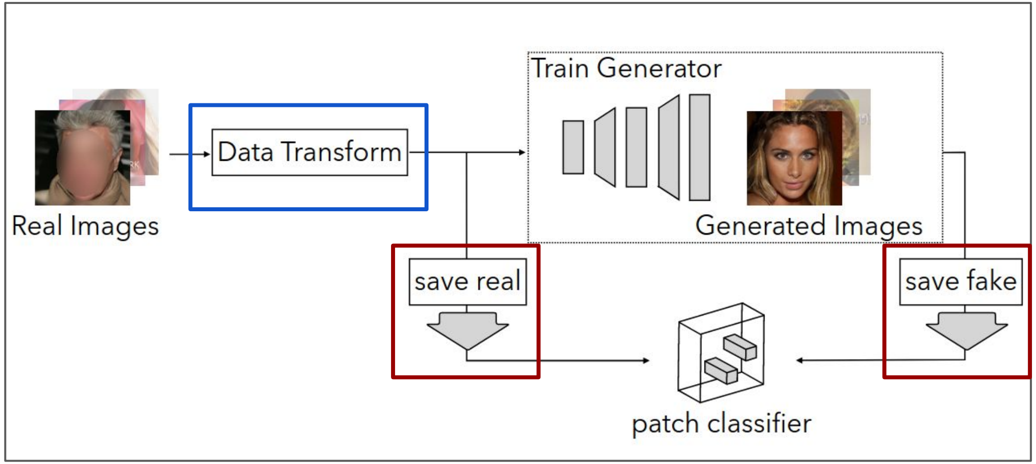



Fig. 2: Dataset processing pipeline

# Generating Dataset

- Fake image generator
  - Generative models: Entirely manipulated images
  - Facial manipulation models: Partially manipulated images
- Real images
  - Celebfaces Attributes (CelebA-HQ)
  - Flickr Faces HQ (FFHQ)

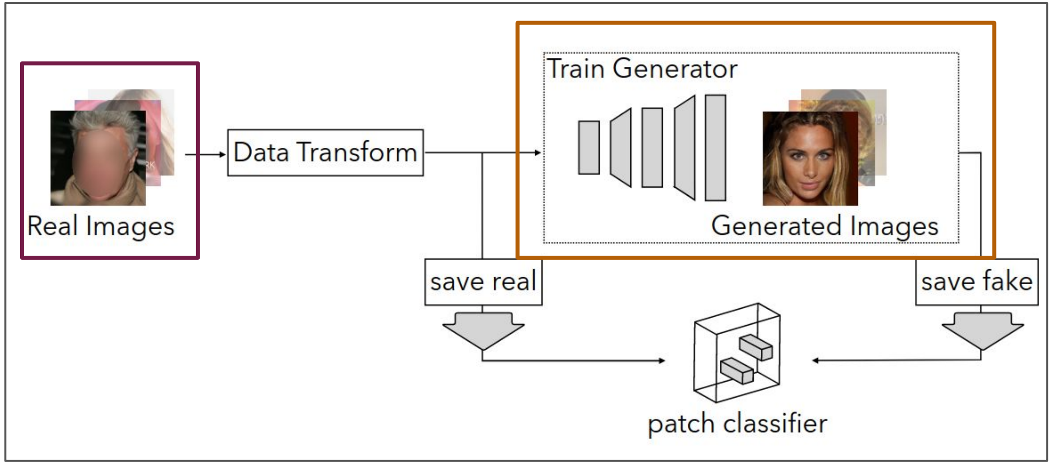


Fig. 2: Dataset generation

# Experiments

- Classification via patches
  - Progressive GAN (PGAN), StyleGAN (SGAN), StyleGAN2 (SGAN2)
- Facial manipulation
  - Face swap, Deepfake, Neural texture, Face2Face
- Baseline models
  - MesoIncception4
  - Resnet 18
  - Xception
  - Convolution Neural Network (CNN)

# Classification via patches: Resolution

- Training
  - 128px fake images from PGAN
  - 128px real images
- Testing
  - 256 - 1024px images
  - Generated using PGAN using CelebAHQ
- Baseline full models perform worse on unseen resolutions
  - Less generalization

| | Resolution | | | |
|---|---|---|---|---|
| **Model Depth** | **128** | **256** | **512** | **1024** |
| Resnet Layer 1 | 100.0 | 99.99 | 99.60 | 96.95 |
| Xception Block 1 | 100.0 | 100.0 | 99.87 | 98.53 |
| Xception Block 2 | 100.0 | 100.0 | 100.0 | 99.98 |
| Xception Block 3 | 100.0 | 100.0 | 100.0 | 99.92 |
| Xception Block 4 | 100.0 | 100.0 | 99.92 | 99.34 |
| Xception Block 5 | 100.0 | 100.0 | 98.90 | 91.18 |
| [2] MesoInception4 | 100.0 | 99.59 | 98.15 | 87.00 |
| [13] Resnet-18 | 99.99 | 96.85 | 91.75 | 80.17 |
| [6] Xception | 100.0 | 99.94 | 99.84 | 97.28 |
| [33] CNN (p=0.1) | 100.0 | 99.99 | 99.97 | 99.78 |
| [33] CNN (p=0.5) | 100.0 | 100.0 | 99.99 | 99.83 |

Table 1: Average precision for different image resolutions

# Classification via patches: Model Seed

- Training
  - 128px fake images from PGAN
  - 128px real images
  - Different PGAN model seeds
- Testing
  - Fake images from other generators (SGAN, SGAN2)
- Classification via patches outperform full models
  - Robust to model seed

| | Model Seed | | | |
|---|---|---|---|---|
| Model Depth | 0 | 1 | 2 | 3 |
| Resnet Layer 1 | 100.0 | 100.0 | 100.0 | 100.0 |
| Xception Block 1 | 100.0 | 100.0 | 100.0 | 100.0 |
| Xception Block 2 | 100.0 | 100.0 | 100.0 | 100.0 |
| Xception Block 3 | 100.0 | 100.0 | 100.0 | 100.0 |
| Xception Block 4 | 100.0 | 100.0 | 100.0 | 100.0 |
| Xception Block 5 | 100.0 | 100.0 | 100.0 | 100.0 |
| [2] MesoInception4 | 100.0 | 99.99 | 99.82 | 99.95 |
| [13] Resnet-18 | 99.99 | 98.41 | 95.20 | 95.02 |
| [6] Xception | 100.0 | 100.0 | 99.99 | 100.0 |
| [33] CNN (p=0.1) | 100.0 | 100.0 | 100.0 | 100.0 |
| [33] CNN (p=0.5) | 100.0 | 100.0 | 100.0 | 100.0 |

Table 1: Average precision for different model seed

# Classification via patches: Generator Architecture

- Training
  - Random samples from PGAN
  - Reprojected images PGAN images
- Testing
  - Other generator architectures
    - SGAN
    - **G**enerative F**low** (Glow)
    - Gaussian Mixture Model (GMM)
- Outperform complete models
  - Easiest generalization: SGAN
    - Similar architectures
  - Except Glow

| | Architectures | | | |
|---|---|---|---|---|
| **Model** | **PGAN** | **SGAN** | **Glow*** | **GMM** |
| Resnet Layer 1 | 100.0 | 97.22 | 72.80 | 80.69 |
| Xception Block 1 | 100.0 | 98.68 | 95.48 | 76.21 |
| Xception Block 2 | 100.0 | 99.99 | 67.49 | **91.38** |
| Xception Block 3 | 100.0 | **100.0** | 74.98 | 80.96 |
| Xception Block 4 | 100.0 | 99.99 | 66.79 | 42.82 |
| Xception Block 5 | 100.0 | 100.0 | 60.44 | 48.92 |
| [2] MesoInception4 | 100.0 | 97.90 | 49.72 | 45.98 |
| [13] Resnet-18 | 100.0 | 64.80 | 47.06 | 54.69 |
| [6] Xception | 100.0 | 99.75 | 55.85 | 40.98 |
| [33] CNN (p=0.1) | 100.0 | 98.41 | 90.46 | 50.65 |
| [33] CNN (p=0.5) | 100.0 | 97.34 | **97.32** | 73.33 |

Table 2: Average precision for different generator architectures

# Classification via patches: Datasets

- Training
  - Random samples from PGAN
  - Reprojected images PGAN images
  - CelebAHQ images
- Testing
  - FFHQ real images
  - FFHQ Fake images using
    - PGAN, SGAN, SGAN2
- Outperform complete baseline models
  - FFHQ has greater diversity

| | FFHQ dataset | | |
|---|---|---|---|
| **Model** | **PGAN** | **SGAN** | **SGAN2** |
| Resnet Layer 1 | 99.81 | 72.91 | 71.81 |
| Xception Block 1 | 99.68 | 81.35 | 77.40 |
| Xception Block 2 | **100.0** | 90.12 | 90.85 |
| Xception Block 3 | 100.0 | 92.91 | **91.45** |
| Xception Block 4 | 100.0 | **95.85** | 90.62 |
| Xception Block 5 | 100.0 | 93.09 | 89.08 |
| [2] MesoInception4 | 98.71 | 80.57 | 71.27 |
| [13] Resnet-18 | 79.20 | 51.15 | 52.37 |
| [6] Xception | 99.94 | 85.69 | 74.33 |
| [33] CNN (p=0.1) | 99.95 | 90.48 | 85.27 |
| [33] CNN (p=0.5) | 99.93 | 88.98 | 84.58 |

Table 2: Average precision across FFHQ dataset

# Classification via patches: Summary

- Outperforms baseline models across
  - Different image resolutions
  - Generator seeds
  - Generator architectures
  - Different datasets

=> Model generalizes concepts for classification for synthesized images.

# Facial Manipulation

- Blend content from two images
  - Portion if image is manipulated
- Datasets
  - Face swap, Deepfake, Neural texture, Face2Face
- Train on one of dataset
- Test for generalization on others
  - Best generalization: Face2Face
  - Least generalization: FaceSwap

| | Train on Face2Face | | | |
|---|---|---|---|---|
| Model Depth | DF | NT | F2F | FS |
| Resnet Layer 1 | **84.39** | 79.72 | 97.66 | 60.53 |
| Xception Block 1 | 77.65 | **80.88** | 93.84 | 61.62 |
| Xception Block 2 | 84.04 | 79.51 | 97.40 | 63.21 |
| Xception Block 3 | 76.10 | 74.77 | 97.33 | 63.10 |
| Xception Block 4 | 67.18 | 61.72 | 97.19 | 63.04 |
| Xception Block 5 | 81.25 | 61.91 | 96.45 | 55.15 |
| [2] MesoInception4 | 67.53 | 55.17 | 92.27 | 54.06 |
| [13] Resnet-18 | 55.43 | 52.57 | 93.27 | 53.39 |
| [6] Xception | 66.12 | 56.07 | 97.41 | 53.15 |
| [33] CNN (p=0.1) | 65.76 | 64.81 | 98.40 | 59.48 |
| [33] CNN (p=0.5) | 65.43 | 60.36 | 97.94 | **63.52** |

Table 2: Average precision across FFHQ dataset

# Generalization in Facial Manipulation

- Classifiers use facial features to classify
  - Without explicit supervision
- Predominantly use mouth
  - Eyes or nose as a secondary feature

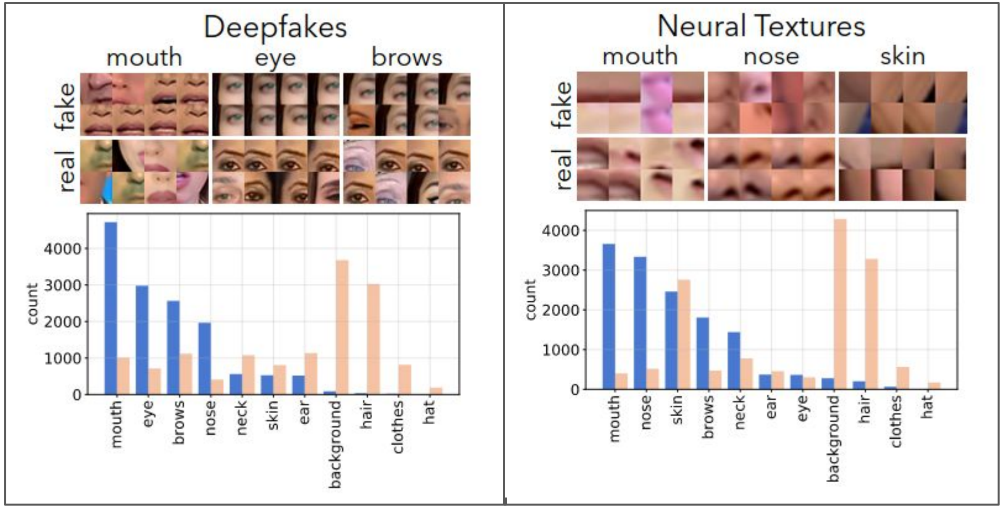



Table 2: Average precision across FFHQ dataset

# Summary

- Shallow models can classify fake images
  - Limited receptive field
  - Using local features
  - Significantly less parameters
- Captures imperfections of generators
  - Local semantics instead of global
- Patch classifier can
  - localize regions of manipulation
  - Imperfective regions of generators

- Applications
  - Help to look for potential manipulations
  - Better navigate falsified content
  - Factual verification of content
- Challenges of fake images
  - Increasingly becoming easier to generate images
  - Generative models can exploit learnings of classification models

[https://chail.github.io/patch-forensics/](https://chail.github.io/patch-forensics/)