Pixel Based Complex Background Subtraction for Video Compositing

Sunny He[1] Yannis Karakozis[2]

[1] Princeton University, Department of Electrical Engineering, [2]Princeton University, Department of Computer Science

Emails: slhe@princeton.edu ick@princeton.edu

Background and Motivation

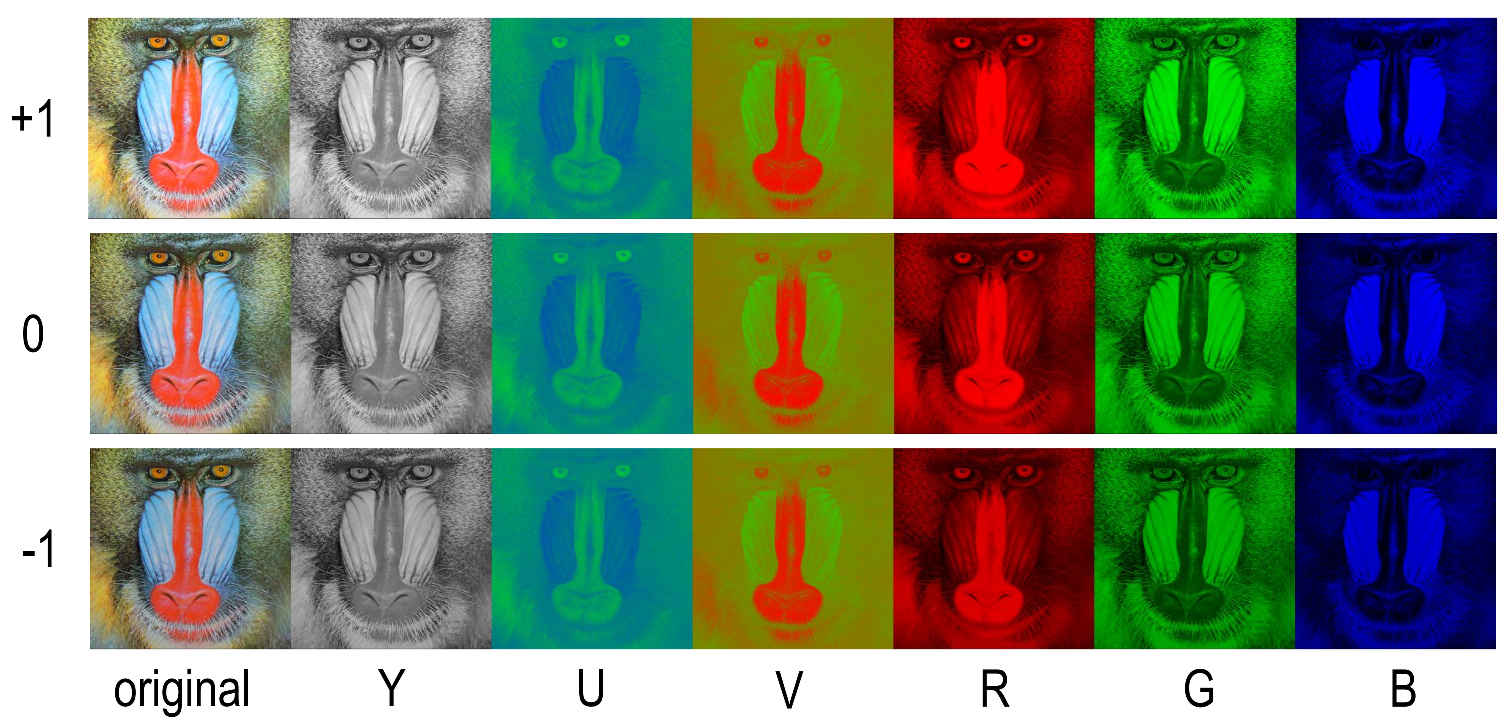
- Rotoscoping is process of cutting out foreground from background
- Important step of visual effects pipelines
  - Overlay studio footage of actors on background footage
  - Combine live-action and computer generated footage
- Currently professional rotoscoping is largely done by hand
  - Time consuming and expensive
- Chroma-key (Green screen) techniques are difficult to implement
  - Extensive set preparation required

Related Work

- Problem has been subject of research in different contexts
- Pixel color models
  - Develop statistical models for background vs. foreground
  - Mixture of Gaussian (MOG) model (Zivkovic 2004)
- Trimap based segmentation
  - Pre-label areas of known foreground, background
  - "Video Matting of Complex Scenes" (Chuang 2002)
- Can we strike a balance in the amount of info given to the algorithm?
  - Pre-labeling with trimaps requires a lot of manual work
  - Possible to pre-train pixel color models if background is known?

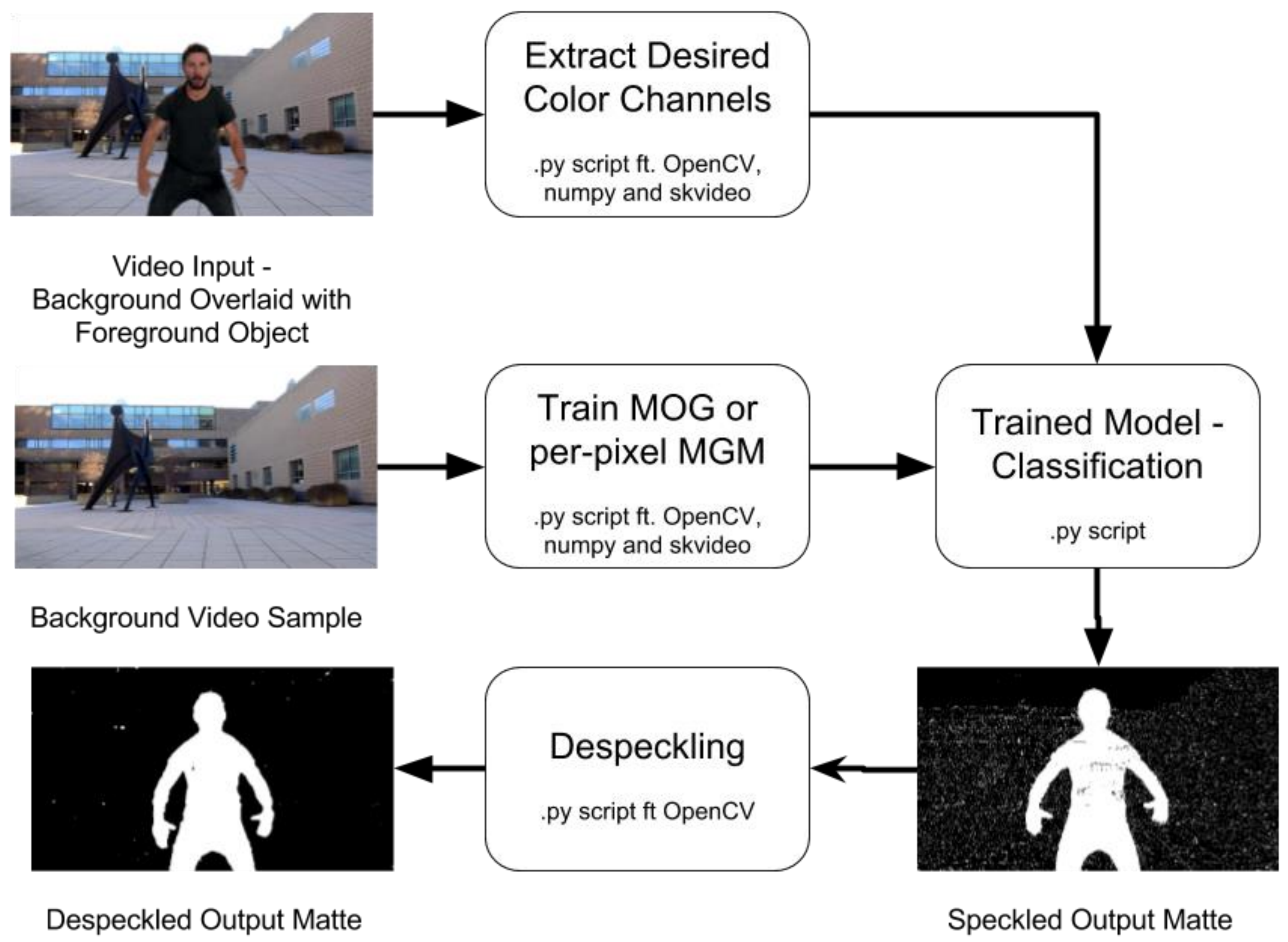
Color Space Transformation

- Existing pixel color model segmentation struggles with lighting changes
- RGB values differ greatly as lighting changes
- Experiment with YUV color space
  - Luma (Y) channel isolates perceived brightness
  - Reduces covariance between components under lighting change



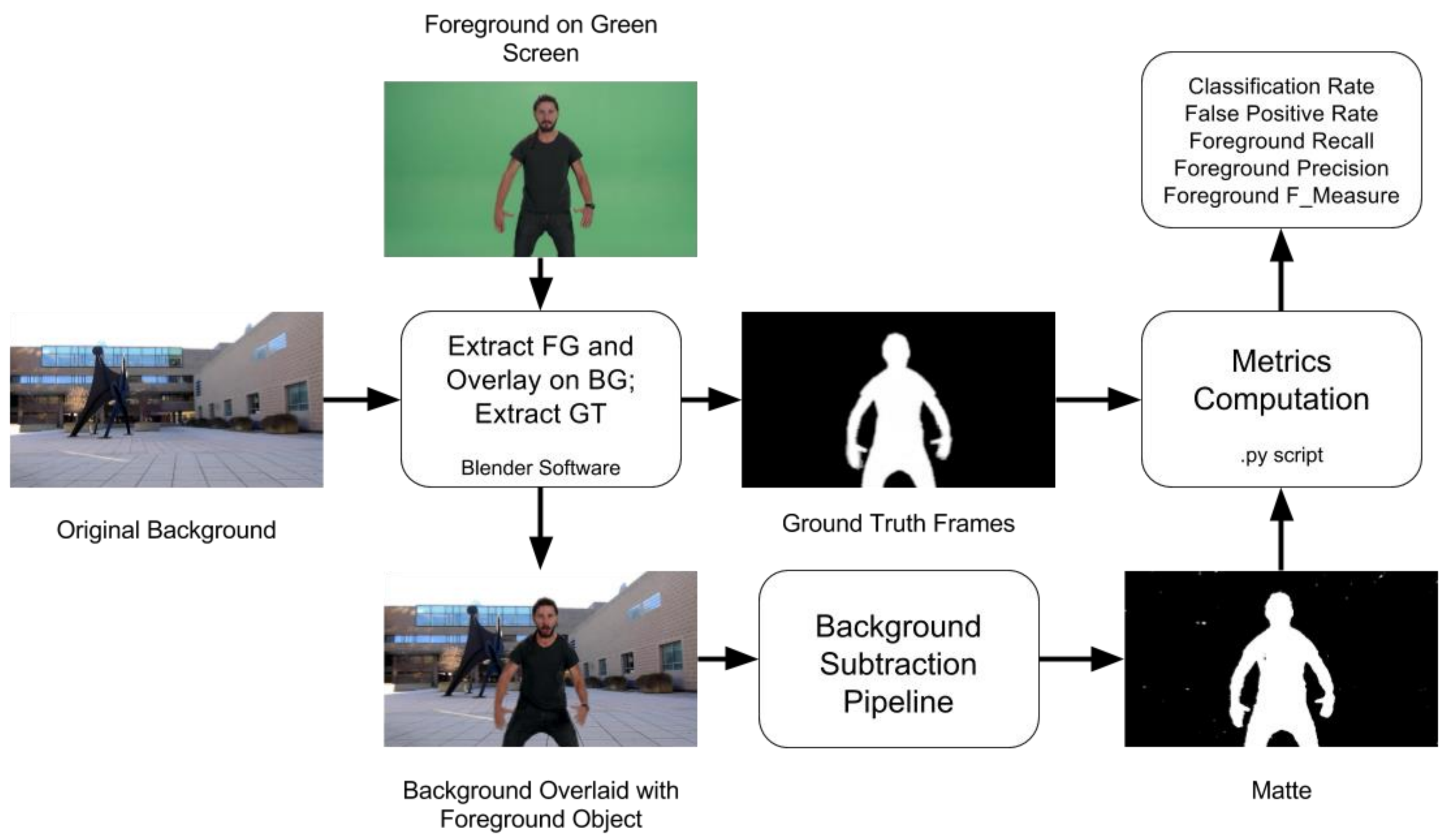
Background Subtraction Pipeline

- Implemented in Python with NumPy, OpenCV, SkVideo
  - MOG model based on code from Zivkovic 2004
  - Per-pixel Multivariate Gaussian Model (MGM) on color channels
- Short sample footage of static background provided to train models
- Despeckling post-processing on segmentation output
- Final result is binary matte



Testing Pipeline

- Generate composite scenes with known foreground and background
- Compare output segmentation with known foreground segmentation
- Tested with three different foregrounds, two different backgrounds
  - Smooth vs. fast motion, indoor vs. outdoor environments
  - Single vs. multiple objects



Performance Evaluation

- Evaluated performance of algorithms with different parameters
- Six different input foreground/background combinations
- Using YUV color space produces higher accuracy and F-score
- Median filter greatly improves performance
  - Larger median filters and more iterations does not significantly improve performance

Color Space Performance Comparison

All metrics are means over all frames of the video		Average Performance over All 6 Videos				
		Accuracy	False FG	FG Recall	FG Precision	FG F-Score
	Color Space and Channels Used					
MOG	RGB	97.46%	0.58%	85.15%	91.73%	87.38%
	RGB - No LUMA	95.90%	3.95%	80.66%	83.82%	81.55%
	YUV	97.86%	1.10%	89.05%	90.61%	89.16%
	UV	95.93%	0.41%	75.62%	92.05%	82.07%
Per Pixel MGM	RGB	96.18%	4.16%	98.04%	78.79%	86.23%
	RGB - No LUMA	96.83%	0.65%	83.90%	89.85%	86.56%
	YUV	97.51%	2.92%	97.61%	83.08%	89.24%
	UV	96.79%	0.41%	82.36%	92.36%	86.87%

Despeckling Filter Performance Comparison

	Median Filter Application Iterations					
Filter Size	Zivkovic MOG - YUV			Per Pixel MGM - YUV		
	1	10	100	1	10	100
0 (no filtering)	66.13%			90.89%		
3	81.70%	83.42%	83.43%	97.23%	98.19%	98.21%
5	83.29%	83.27%	82.57%	92.29%	98.52%	98.51%
7	83.28%	82.60%	80.85%	98.36%	98.39%	97.07%

Results and Future Work

- Successfully demonstrated background subtraction
- Color Space
  - YUV does improve performance
  - However, luminance still plays a role (RGB better than UV)
  - YUV also has trouble with grayscale subjects/backgrounds
- Model comparisons
  - Pre-trained per-pixel model performs better than running model
  - Strong assumption of a static background with little changes
- Despeckling
  - Median filter improves performance drastically
  - Quantitatively, size/iterations has a much smaller effect
    - However, qualitatively seems better
- Future Work
  - All models used Gaussian distributions
    - Perhaps different distributions/statistical tests are better
  - Moving backgrounds
    - Optical flow methods
  - Other despeckling filters/methods