

Distinguishing Authentic from AI-Generated Explosions using Spatiotemporal Dynamics

Sarah Barrington and Hany Farid
University of California, Berkeley
Berkeley CA, USA

sbarrington@berkeley.edu and hfarid@berkeley.edu

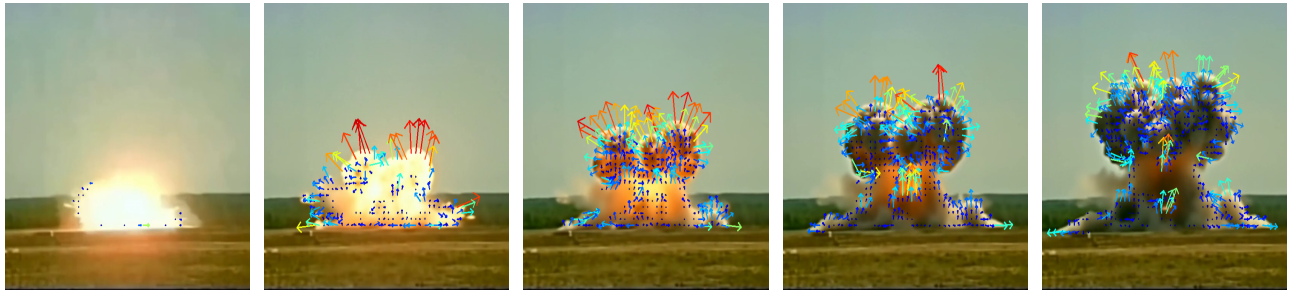


Figure 1. Five frames of an AI-generated explosion annotated with the estimated optical flow (the vectors denote motion direction and magnitude encoded by color with red denoting large motion and blue denoting small motion).

Abstract

Understanding and responding to world events during military conflicts has always been challenging. This has been heightened in the age of deepfakes. We explore the problem of authenticating conflict-zone footage depicting explosions. We find that otherwise high-performing forensic techniques struggle to reliably distinguish between authentic and AI-generated videos. We hypothesize that this is because the spatiotemporal dynamics of explosions are highly distinct, falling outside the videos used to train forensic classifiers. By combining domain-specific classifiers – trained exclusively on explosions – and physics-based modeling, we show that accuracy on detecting explosive content jumps from near chance to above 90%.

1. Introduction

In May of 2023, a fake image purporting to show an explosion at the Pentagon went viral online just as the U.S. stock market was opening. In the ensuing confusion, the S&P 500 index temporarily dropped by some \$500 billion dollars before recovering [5]. This incident highlighted the risks that emerge when deepfakes meet the virality of social media.

In the intervening years, the quality and realism of AI-generated images [20] and voices [3] have improved

tremendously. More recently, new techniques have emerged to create highly realistic videos (e.g., Sora, Veo, and Runway). These AI-generated images and videos are being weaponized in the form of non-consensual intimate imagery, child sexual abuse imagery, small- to large-scale fraud, and disinformation campaigns [7]. A particularly troubling abuse of generative AI has been in sowing confusion in conflict and war zones. Throughout the Gaza war, for example, social media has been littered with all forms of fake images and videos, adding confusion to an already complex and emotionally charged conflict [16]. In addition, the mere presence of these fakes has cast a shadow of doubt over authentic content [11]. And, more recently, a fake video went viral purporting to show a massive nuclear explosion caused by US strikes in the city of Arak, Iran [1].

While the field of image and video forensics is progressing, accurate, scalable, and generalizable detection of AI-generated videos is proving somewhat challenging. In the recent SAFE Synthetic Video Detection Challenge 2025, 12 teams competed in a blind evaluation over 90 days [14]. The first of two tasks focused on identifying video clips (with no audio) from 13 video-generation models and 21 video-hosting services. These AI-generated and authentic videos did not undergo any post-processing operations. The second task assessed detection accuracy under 14 common post-processing applications including downscaling, additive noise, color correction, text overlay, etc.

Averaged across the top-five performing teams, for the first task the true positive rate (correctly identifying an AI-generated video) was 78.8% with an average true negative rate (correctly identifying an authentic video) of 86.6%. On the more challenging second task the true positive and negative rate dropped to 70.4% and 71.0%. The top-performing team (as determined by AUC averaged over both tasks) achieved a true positive and negative rate of 75% and 85%. Although the top performers struggled to generalize to post-processing artifacts, they were more successful in generalizing across generative-AI models not present during training.

Simultaneously, a recent evaluation by Runway AI reported that over 90% of viewers could not reliably distinguish their state-of-the-art AI-generated videos from authentic footage [24]. Overall, there are significant challenges to computationally and perceptually distinguishing authentic from AI-generated videos at an accuracy that is appropriate in high-stakes scenarios like validating content from conflict and war zones.

Adding to this challenge is that videos of explosions are semantically anomalous relative to typical datasets because, while explosions are common in video games and movies, these are typically computer-generated (CGI) while real-world explosions are far rarer (there are several datasets that capture fire and smoke [18, 27], however this type of continuous combustion and persistent plume is distinct from our modeling of transient plume dynamics from a single explosive event). This type of out-of-domain classification is known to pose additional challenges to deepfake detectors. We begin by confirming this by evaluating two state-of-the-art deepfake detectors based on spatial and spatiotemporal embeddings, each of which achieve 95.4% and 94.5% average detection accuracies on in-domain testing data. Detection accuracies on authentic and AI-generated (hereafter referred to as synthetic) videos of explosions, however, drop to 52.3% and 54.6%. We then show that after retraining these detectors on our dataset, detection accuracies improve to 77.8% and 91.7%.

We hypothesize that retraining does not fully close the accuracy gap because explosions are both spatially and temporally highly complex and may not be as easily captured using embeddings trained on more “natural” motion. To further improve detection, we explicitly model the photometric and dynamic properties of explosions, and show that an ensemble detector combining spatiotemporal embeddings and these physics-based properties achieves 94.4% accuracy. We conclude with thoughts on how this hybrid approach can be further refined to enable authentication of critical content emerging from global conflicts and wars.

2. Methods

We begin by describing the various video datasets used in our analyses, followed by a three-part methodology that de-

scribes our quantification of explosions using spatial embeddings, spatiotemporal embeddings, and spatiotemporal dynamic motion and photometric properties.

2.1. Datasets

Baseline videos: The video datasets that we used to retrain and evaluate state-of-the-art benchmark deepfake detectors (Sections 2.2 and 2.3) include 50,000 natural videos [23] and 50,000 AI-generated videos, spanning five different models [28] (see <https://github.com/ChristianInterno/ReStraV>). We consider only the first two seconds of each video to ensure consistency with the baseline approaches, and because much explosive activity happens in this time frame.

Authentic explosions: We were provided a set of 82 videos of controlled explosions from the FBI’s Terrorist Explosive Device Analytical Center (TEDAC). These videos included a range of explosions and materials, from small at-home explosive materials to large-scale detonations. These videos were augmented with 10 YouTube videos of military missile tests. These 92 videos were manually filtered with the following criteria: (1) stable camera; (2) standard video rate (i.e., no slow motion); and (3) the post explosion smoke plume is visible for at least two seconds. This filtering yielded only 22 authentic videos. Each video is temporally cropped to the moment just before detonation. These videos varied in spatial resolution and frame rate, an issue that we address below.

Synthetic explosions: Each of the 22 authentic videos was auto-captioned using the Gemini image vision-language model¹ to yield captions like:

“A static wide shot on a bright, sunny day captures a sudden ground explosion in a wide, dry, grassy field. A dense, light brown and white plume of dust and smoke erupts from the ground, rapidly rising and expanding upwards before slowly dissipating and drifting slightly to the right. In the background, rolling hills covered with dense, autumnal trees stretch across the horizon beneath a clear blue sky. A small, dark, shed-like structure is partially visible near the tree line in the midground.”

Each caption along with the first frame of each corresponding video is used as input into one of four different video

¹For each video, Gemini is prompted with the following: “What is happening in this video? Make sure your response is a prompt that is suitably descriptive for a t2v engine, ignoring any additive post-processing video touches such as graphics or text. Also ensure you comment on the surrounding environment and the camera being stationary. If you see an explosion, ensure that you state there is only one explosion.”

Authentic



Runway4



Sora2



Veo3-1



Figure 2. Select frames from an authentic video (top row) and three AI-generated videos created using image-to-video generation from Runway4, Sora2, and Veo3-1. Each video is generated from the first frame of the authentic video and the prompt “A wide, stationary shot captures a sudden, powerful explosion in a dry, grassy field. A large, yellowish-orange cloud of smoke billows rapidly upwards and outwards, then slowly dissipates, leaving a white, wispy smoke trailing along the ground. In the background, a dense line of trees displays autumn colors, with distant hills visible under a clear blue sky, marked by two faint airplane contrails. A small, dark shed-like structure is partially visible near the treeline behind the explosion site.”

generators: Runway4, Sora2, Veo3-1, and Veo3 (each accessed through the API of Pollo.ai). These generators support a set number of fixed aspect ratios. For input frames that do not match these ratios, a closest compatible aspect ratio is selected to which the input frame is cropped. Due to differences across video generators, the synthetic videos ranged in length from 5 to 8 seconds. As with the authentic videos, these videos varied in spatial resolution and frame rate, an issue that we address below. In total, we requested $176 = 22 \times 4 \times 2$ synthetic videos, from 22 authentic sources, 4 generators, and 2 samples per generation. Due to sporadic API failures and content moderation limits, we collected a total of 99 synthetic videos. The first two seconds of these videos comprise 1,652 authentic and 5,100 synthetic video frames.

2.2. Spatial Embeddings

Training classifiers on spatial embeddings extracted by pre-trained transformer architectures have proven highly effective in the field of deepfake detection across video [4, 21], image [19], and audio [2]. We therefore propose a baseline classification model using embeddings from DINOv2, a

state-of-the-art vision transformer. Videos are resampled to 24 frames-per-second and spatial embeddings are extracted using the DINOv2 ViT-S/14 visual encoder [22]. The “classify” (CLS) token from the final transformer block is extracted at the frame-level, resulting in a 384-dimensional embedding per frame, which serves as the input for downstream classification.

2.3. Spatiotemporal Embeddings

Spatial embeddings capture static, frame-level features but not temporal features. To this end, we employ the spatiotemporal approach termed “perceptual straightening” [12]. This approach is motivated by the observation that natural videos exhibit geometric regularities in a neural representation [10].

This analysis begins by extracting, per video frame, the same spatial DINOv2 CLS token as above from the resampled videos (24 frames-per-second). This embedding is augmented with the 196 (14×14) patch-wise embeddings extracted from the transformer block. These embeddings are flattened and concatenated to form a single feature vector (75,648-D) per video frame. The image sequence is then

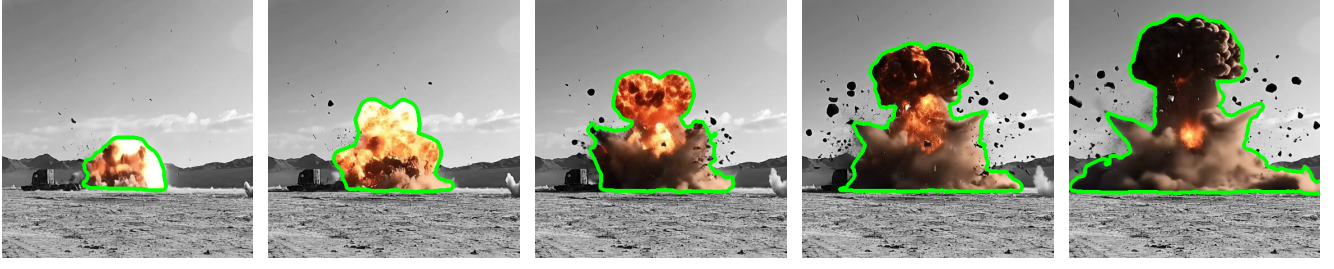


Figure 3. Select frames of frame-based segmentation to isolate an explosion. For purposes of visualization, the region outside the explosion is desaturated.

quantified by computing temporal first- and second-order derivatives of these features. In particular, denoting the embedding at frame i as $\vec{z}_i$, the first-order temporal derivative is given by:

$$\Delta \vec{z}_i = \vec{z}_{i+1} - \vec{z}_i. \quad (1)$$

The magnitude of which is $\|\Delta \vec{z}_i\|_2$. The temporal second derivative (curvature) is given by:

$$\theta_i = \arccos \left(\frac{\Delta \vec{z}_i \cdot \Delta \vec{z}_{i+1}}{\|\Delta \vec{z}_i\|_2 \|\Delta \vec{z}_{i+1}\|_2} \right) \quad (2)$$

Summary statistics (mean, variance, min, max) are extracted from these temporal first- and second-order derivatives. This yields an 8-D feature per video. Notably, the distributed code² differs from this written description in that the final extracted features consist of these eight summary statistics along side 13 additional features including $\vec{z}_i$, $i \in [1, 7]$ and θ_i , $i \in [1, 6]$. In comparing our results to this baseline approach, we use the authors’ 21-D representation, but when we train our custom model, we revert to the 8-D representation.

2.4. Spatiotemporal Dynamics

We hypothesize that the spatial and spatiotemporal embeddings will be able to capture some differences between authentic and synthetic explosions. We also hypothesize that these generic embeddings will benefit from more physically-based analyses that capture the complex dynamics of explosions.

2.4.1. Segmentation

Because we are only interested in the temporal dynamics of the explosion and not the background, we segment each frame of a video sequence to isolate the explosion and corresponding smoke plume. Each video frame is first segmented using the grounding DINO [17] (tiny) model for zero-shot object detection, prompted with the terms “explosion”, “plume”, and “smoke”. This produces a bounding box with highest likelihood of containing an explosion

(we specify a confidence threshold of 0.35). The bounding box is then passed to the segment anything model (SAM) ViT-H vision transformer [15], which yields a segmentation mask, Figure 3.

2.4.2. Motion

An image sequence (video) is denoted as $f(x(t), y(t), t)$, where $x(t)$ and $y(t)$ are the spatial parameters and t is the temporal parameter. Given such an image sequence, our task is to estimate the amount of motion (optical flow) at each point in the image at each moment in time [26]. For a given instant in space and time, we require an estimate of velocity denoted as the 2D motion vector $\vec{v} = (v_x \ v_y)$, where v_x and v_y denote the horizontal and vertical components of the velocity vector $\vec{v}$.

In order to estimate motion, an assumption of brightness constancy is made. That is, it is assumed that as a small surface patch is moving, its brightness value remains unchanged. This constraint can be expressed with the following partial differential equation:

$$\frac{\partial f(x(t), y(t), t)}{\partial t} = 0. \quad (3)$$

This constraint holds for each point in space and time. Expanding this constraint according to the chain rule yields:

$$\frac{\partial f}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial t} + \frac{\partial f}{\partial t} = 0, \quad (4)$$

where the partials of the spatial parameters x and y with respect to time correspond to the velocity components:

$$f_x v_x + f_y v_y + f_t = 0. \quad (5)$$

The subscripts on the function f denote partial derivatives. Note again that this constraint holds for each point in space and time but that for notational simplicity the spatial/temporal parameters are dropped. This transformed brightness constancy constraint is rewritten by packing together the partial derivatives and velocity:

$$\begin{aligned} (f_x \ f_y) \begin{pmatrix} v_x \\ v_y \end{pmatrix} + f_t &= 0 \\ \vec{f}_s^t \vec{v} + f_t &= 0. \end{aligned} \quad (6)$$

²<https://github.com/ChristianInterno/ReStraV>

The space/time derivatives $\vec{f}_s$ and f_t are measured quantities, leaving us with a single constraint in two unknowns (the 2D velocity vector, $\vec{v}$). The constraint can be solved by assuming that the motion is locally similar, and integrating this constraint over a local image neighborhood. A least-squares error function takes the form:

$$E(\vec{v}) = \left[\sum_{x,y} \vec{f}_s^t \vec{v} + \sum_{x,y} f_t \right]^2. \quad (7)$$

To solve for the motion, this error function is differentiated:

$$\frac{\partial E(\vec{v})}{\partial \vec{v}} = 2 \sum \vec{f}_s \vec{f}_s^t \vec{v} + 2 \sum \vec{f}_s f_t. \quad (8)$$

Setting equal to zero and recombining the terms into matrix form yields:

$$\begin{pmatrix} \sum f_x^2 & \sum f_x f_y \\ \sum f_x f_y & \sum f_y^2 \end{pmatrix} \begin{pmatrix} v_x \\ v_y \end{pmatrix} = - \begin{pmatrix} \sum f_x f_t \\ \sum f_y f_t \end{pmatrix}$$

$$M \vec{v} = -\vec{b}. \quad (9)$$

If the matrix M is invertible (full rank), then the velocity can be estimated by simply left multiplying by the inverse matrix:

$$\vec{v} = -M^{-1} \vec{b} \quad (10)$$

Generally speaking the matrix is not invertible (i.e., rank deficient), when the intensity variation in a local image neighborhood varies only one-dimensionally (e.g., $f_x = 0$ or $f_y = 0$) or zero-dimensionally ($f_x = 0$ and $f_y = 0$). These singularities are sometimes referred to as the aperture and blank wall problem. The motion at such points simply can not be estimated.

We employ a multi-scale, optical-flow estimation [8] that yields spatially and temporally dense motion fields (Figure 1). Due to differences in video frame rates and spatial resolution across our dataset, we normalize the estimated motion (v_x, v_y), specified in pixels/frame, to adjust for these differences. Each component is multiplied by the video's frame rate (fps) to convert to pixels/second, and then divided by the frame width (W) and height (H) to express motion as a proportion of the frame rather than in pixels:

$$\tilde{v}_x = \frac{fps}{W} v_x \quad \text{and} \quad \tilde{v}_y = \frac{fps}{H} v_y. \quad (11)$$

All pixels outside of the per-frame segmentation mask (Section 2.4.1) are zeroed-out and ignored in later calculations.

The final dataset described in Section 2.1 is narrowed down from the full set of videos by removing any videos where the segmentation or optical flow estimation clearly failed.

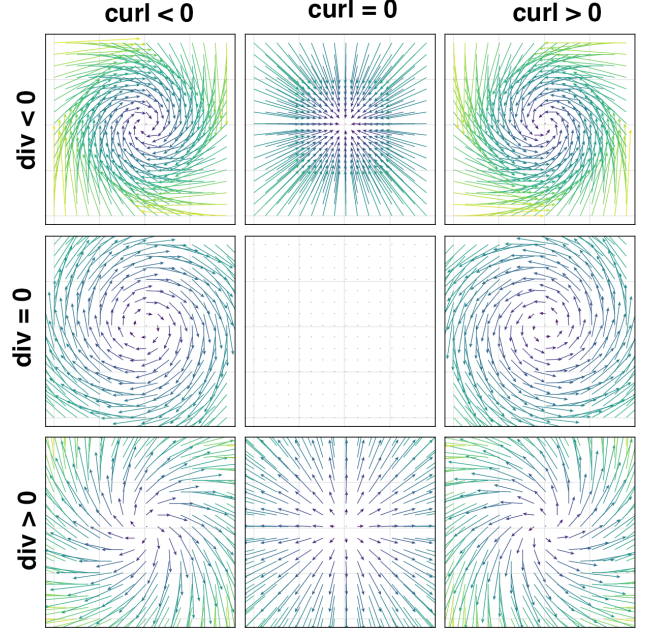


Figure 4. Different local motion fields for positive, negative, and zero values of div and curl. For zero curl, positive and negative values of div correspond to a local expansion and contraction (center column). For zero div, positive and negative values of curl correspond to counter-clockwise and clockwise rotation (center row). Non-zero div and curl correspond to combinations of these patterns (four corners).

Div, grad, curl: We quantify the local spatiotemporal geometric properties of the motion field using standard measures from fluid mechanics: the divergence (div), gradient (grad), and curl of a vector-valued motion field [25]. While typically defined in three dimensions, we necessarily use a 2D formulation.

As shown in Figure 4, div and curl capture locally structured rotation and expansion/contraction, while the grad captures the motion's local linear behavior. In 2D Cartesian coordinates, div, grad, and curl (∇) are defined as:

$$\text{div}(\tilde{v}_x, \tilde{v}_y) = \frac{\partial \tilde{v}_x}{\partial x} + \frac{\partial \tilde{v}_y}{\partial y} \quad (12)$$

$$\text{curl}(\tilde{v}_x, \tilde{v}_y) = \frac{\partial \tilde{v}_y}{\partial x} - \frac{\partial \tilde{v}_x}{\partial y} \quad (13)$$

$$\nabla(\tilde{v}_x, \tilde{v}_y) = \begin{pmatrix} \frac{\partial \tilde{v}_x}{\partial x} & \frac{\partial \tilde{v}_x}{\partial y} \\ \frac{\partial \tilde{v}_y}{\partial x} & \frac{\partial \tilde{v}_y}{\partial y} \end{pmatrix} \quad (14)$$

These three measurements are computed per video frame over local spatial neighbors corresponding to square windows with of size 20% width of the bounding box containing the entire segmentation mask. Summary statistics (mean, standard deviation, skew, and kurtosis) from each

	feature(s)	dataset	model	authentic accuracy (%)	synthetic accuracy (%)	F1-score
(1)	spatial	baseline	baseline	95.4	95.4	0.95
(2)	spatiotemporal	baseline	baseline	93.3	95.6	0.95
(3)	spatial	explosions	baseline	50.0	54.6	0.51
(4)	spatiotemporal	explosions	baseline	86.4	22.7	0.66
(5)	spatial + spatiotemporal	explosions	baseline	59.1	59.1	0.59
(6)	spatial	explosions	custom	77.8	77.8	0.70
(7)	spatiotemporal	explosions	custom	88.9	94.4	0.91
(8)	dynamics	explosions	custom	83.3	83.3	0.80
(9)	spatial + spatiotemporal	explosions	custom	89.5	89.5	0.88
(10)	spatial + dynamics	explosions	custom	88.9	88.9	0.85
(11)	spatiotemporal + dynamics	explosions	custom	94.4	94.4	0.93
(12)	spatial + spatiotemporal + dynamics	explosions	custom	89.5	89.5	0.86

Table 1. Per-class, cross-validation, detection accuracy for different features, datasets, and models. The spatial and spatiotemporal features are extracted from static and temporal DINOv2 embeddings, while the dynamic features are extracted from temporal photometric and geometric properties of explosions. The **bold-faced values** correspond to the best performing features on videos of explosions. The F1-score is defined as $2 \times (\text{precision} \times \text{recall}) / (\text{precision} + \text{recall})$, where precision corresponds to the proportion of predicted synthetic videos that are synthetic, and recall corresponds to the proportion of synthetic videos that are correctly detected.

video frame are computed from the scalar-valued div and curl, the div and curl magnitude, and the magnitude of the grad, yielding a 20-D feature vector.

2.4.3. Photometric

In addition to modeling the geometric properties of explosions, we also model the temporal dynamics of its photometric properties. Specifically, we measure the temporal changes in hue and brightness of the segmented explosion and smoke plume. Each frame is converted to HSV color space. Hue (H) is represented in degrees (0 - 180) and brightness (V) in the intensity range [0, 255].

Summary statistics for brightness (mean, standard deviation, skew, and kurtosis) are extracted from the segmented region on each frame. Because hue is an angular quantity, circular summary statistics (mean and standard deviation, skew, and kurtosis) are computed [13]. Temporal changes in hue and brightness are then computed as differences between consecutive frames and standardized to a per-second rate to ensure comparability across videos with differing frame rates. This yields an 8-D feature vector which is combined with the 20-D motion feature vector.

2.5. Classification

2.5.1. Baseline

We trained and evaluated two baseline detectors based on spatial (Section 2.2) and spatiotemporal (Section 2.3) features using a multi-layer perceptron (MLP) per the distributed code. The MLP has an input dimension of 384 (spatial) or 21 (spatiotemporal) and two hidden layers (256/128 (spatial) and 64/32 (spatiotemporal) units) with ReLU activation. Each model is trained for 10 epochs with a batch

size of 32 using the Adam optimizer (BCEWithLogits loss and learning rate $1e-3$). The output layer produces a single logit for binary classification.

For the spatial embeddings, binary classification operates at the frame level, whereas for spatiotemporal embeddings, it operates at the video level. Frame-level categorization is converted to video-level in which a video is classified as synthetic if 57% or more of the frames are classified as synthetic. This threshold is set to balance the true positive rate (TPR) and true negative rate (TNR).

The baseline dataset was evenly split into 50,000 training and 50,000 testing videos (Section 2.1).

2.5.2. Custom

We trained and evaluated three custom detectors based on spatial (Section 2.2) spatiotemporal (Section 2.3), and dynamics (Section 2.4) features using a random forest model with 100 estimators and a maximum depth of 10. This model is chosen over the baseline MLP due to our small video dataset. The output of the spatiotemporal model produces a probability of the video being synthetic, while the output of the frame-level spatial and dynamics models produce a binary classification of a frame as synthetic or authentic. Frame-level classifications for the spatial and dynamics models are converted to video-level predictions using the proportion of frames predicted as synthetic. The threshold for each classifier is selected to balance TPR and TNR, resulting in thresholds of 44% and 53%.

A leave-one-out (22-fold) cross validation method is used to maximize utilization of the 22 real videos. In each fold, 21 real videos and the 21 synthetic counterparts are selected as the training set, with one real video and its syn-



Figure 5. An example of an overly dramatic AI-generated explosion with an atypical mushroom cloud.

thetic counterpart left out for testing (because each authentic video may have more than one synthetic counterpart, the pairing is randomized on each iteration). Because the number of authentic and synthetic frames vary, the training dataset is balanced by downsampled for the frame-based models. Averaged across all 22 folds, we report authentic and synthetic detection accuracy, along with F1-scores.

An ensemble model combines video-level predictions from two or more individual models. The output of each model is averaged and compared to a threshold selected to balance TPR and TNR. These thresholds were: three-way ensemble 0.52; spatial + spatiotemporal 0.73; spatiotemporal + dynamics 0.82; spatial + dynamics 0.73 (Table 1).

3. Results

Shown in the rows (1)-(2) of Table 1 is the baseline model evaluated against the baseline dataset (Section 3). Consistent with the original work, spatial and spatiotemporal features yield classification accuracies around 95.4% and 94.5% (both F1-scores = 0.95). These baseline models, however, completely fail to correctly classify our custom dataset of authentic and synthetic explosions. As seen in rows (3)-(4) of Table 1, the average accuracies drop to 52.3% and 54.6%, where chance performance is 50%.

When a random forest (Section 2.5) is trained on feature sets from our custom dataset, spatial and spatiotemporal accuracies jump to an average of 77.8% and 91.7% (F1-scores = 0.70 and 0.91), with the spatiotemporal feature yielding the best (albeit slightly imbalanced) accuracy (rows (6)-(8) of Table 1). The dynamics model achieves an average accuracy of 83.3% (F1-score 0.80)

Lastly, when these three features are combined with a simple ensemble (Section 2.5), accuracy jumps again with the best performing model – spatiotemporal + dynamics – yielding a balanced accuracy of 94.4%, F1-score = 0.93 (rows (9)-(12) of Table 1). Given the complex and physically constrained temporal properties of explosions, it is perhaps not surprising that the temporal information is more important than the spatial information.

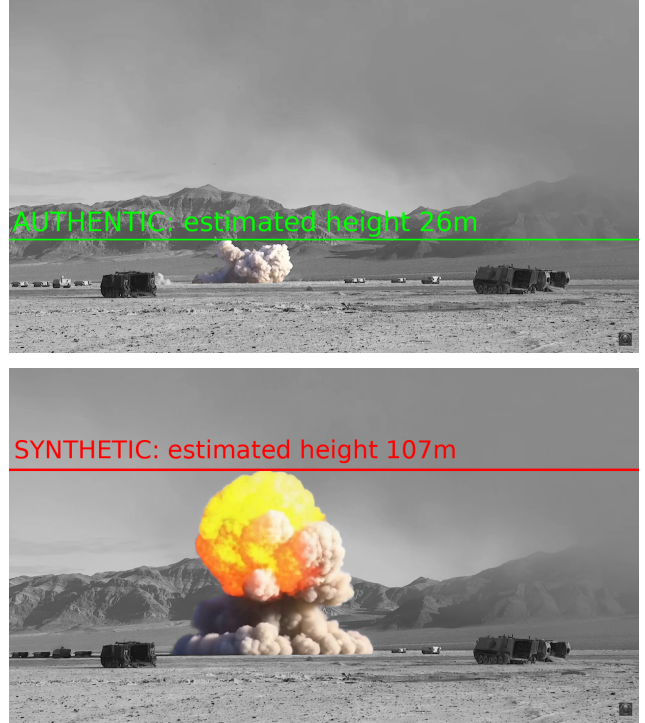


Figure 6. Video frames extracted from the two-second mark of an authentic (top) and synthetic (bottom) video. The synthetic smoke plume, generated by Runway, climbs to a height of 107m (as calibrated from the height of a reference object in the scene) whereas the authentic plume only climbs to a more plausible 26m.

3.1. Qualitative Results

In addition to the spatiotemporal features used to distinguish authentic from synthetic videos of explosions, we have exposed other inconsistencies that, although currently limited due to dataset constraints, may provide insights for future work and for visual investigations.

Mushroom Cloud: Synthetic explosions, particularly those produced by Veo and Runway, frequently show characteristic “mushroom clouds,” Figure 5. While this phenomena can occur in real-world explosions, it is typically associated with fuel-rich detonations or nuclear events, and is comparatively rare in real-world combat situations.

Plume: Smoke plumes from AI-generated explosions, particularly in distant shots, rose both higher and more rapidly than those in real explosions, Figure 6. In order to estimate plume height in real-world units, we require knowledge of a reference object that allows us to convert from pixels to meters. In the case of Figure 6, the armored personnel carrier depicted in the scene has a known height of 2.74m. This approach, while valuable, depends on real-world calibration data which is not always available in our dataset.

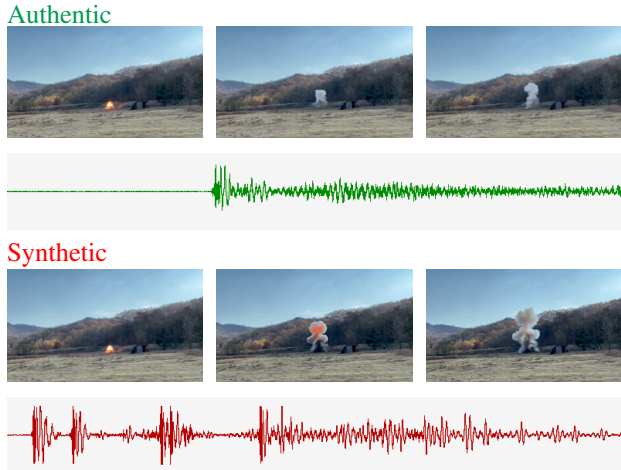


Figure 7. Three video frames from a two-second authentic (top) and synthetic (bottom) video along with the corresponding audio sound waves. From the visual-acoustic offset, we estimate the distance to explosion in the authentic video to be a plausible 229m, whereas the distance to the explosion in the synthetic video is an unrealistic 29m (assuming the speed of sound to be 343m/s).

Visual-Acoustic Offset: Due to differences in the speed of light and sound, a distant explosion is seen before it is heard, Figure 7. This delay is often absent or unrealistically short in synthetic videos. While informative, this cue was not consistently available in our dataset because some synthetic videos do not contain an audio track.

4. Limitations

There are a few limitations that are important to highlight. While there are no limitations in our ability to create synthetic videos of explosions, authentic videos of explosions are harder to come by. This is because unlike more typical videos of people, cars, nature, etc. high-quality, authentic videos of explosions from relevant contexts are less common. This means that the accuracy of our deepfake explosion detection model is yet to be fully validated.

We believe, however, that our results and analysis still provide important insights and guidelines. First, general-purpose models struggle with semantically anomalous content like explosions. Second, for motions that obey photometric and dynamic physical properties, detection models appear to benefit from the introduction of physics-based modeling to sit alongside learned spatiotemporal embeddings. Third, we contribute qualitative observations regarding the real-world plausibility of AI-generated explosions, which we believe could help visual forensic analysts.

Meanwhile, we continue to collaborate with the FBI’s Terrorist Explosive Device Analytical Center (TEDAC) to collect and record more videos across a broad range of detonation sizes.

While our spatial and spatiotemporal analyses are fully automatic, the dynamic analysis requires some manual intervention due to current limitations of image segmentation. We continue to explore new techniques that will remove or minimize the need for this manual intervention.

And lastly, all of our analyses have been constrained to authentic and synthetic videos in which there is no camera motion (zooming or translation) for the two-second window starting at onset of the explosion. While this is a short time window, we are exploring techniques for video stabilization that would enable broader applicability without creating a confound between the camera and scene motion.

5. Discussion

Sir Lonsdale Augustus Hale formally defined the term “fog of war” in 1896 where he describes “the state of ignorance in which commanders frequently find themselves as regards the real strength and position, not only of their foes, but also of their friends” [9]. In our modern world, where nearly every military conflict plays out in real time on 24-hour cable news and social media, today’s fog of war extends beyond military commanders to everyone from journalists to everyday citizens. This fog has gotten even heavier in the age of deepfakes where nearly anyone can create fake visually compelling images and videos.

For example, minutes after the United States’ invasion and capture of Venezuelan President Maduro, social media was awash in fake images and videos purporting to show evidence of the military action, followed by fake videos of Venezuelan’s celebrating in the street [6], all of which led to confusion and uncertainty in the following hours and days.

Despite some limitations in our approach, we believe that bespoke deepfake detectors focused on only one semantic class, like explosions, can play an important role alongside general-purpose detectors. An advantage of this approach is that it can focus on distinctive artifacts without the constraint of having to contend with all types of content and motion. In our case of explosions, for example, we have been able to exploit the complex spatial and spatiotemporal properties of explosions that might otherwise have gone unnoticed. In addition, for situations like ours, with significant barriers to collecting authentic content, a more focused approach with lighter-weight features and machine-learning classifiers may have a better chance of success.

Acknowledgments

We are indebted to Dr. Kirk Yeager, Chief Explosives Scientist at the FBI’s TEDAC for his assistance in acquiring authentic videos of explosions. We would also like to thank Dr. Tom Teschner and Dr. Matt Williams for their help in navigating the field of computational fluid dynamics. This work was funded by the UC Noyce Initiative.

References

- [1] AFP. Mass explosion video is AI-generated, not US attack on Iran. <https://factcheck.afp.com/doc.afp.com.63X6862>, 2025. 1
- [2] Sarah Barrington, Romit Barua, Gautham Koorma, and Hany Farid. Single and multi-speaker cloned voice detection: From perceptual to learned features. In *IEEE International Workshop on Information Forensics and Security*, pages 1–6. IEEE, 2023. 3
- [3] Sarah Barrington, Emily A Cooper, and Hany Farid. People are poorly equipped to detect AI-powered voice clones. *Scientific Reports*, 15(1):11004, 2025. 1
- [4] Matyas Bohacek and Hany Farid. Human action CLIPS: Detecting AI-generated human motion. arXiv:2412.00526, 2024. 3
- [5] Shannon Bond. Fake viral images of an explosion at the Pentagon were probably created by AI. <https://www.npr.org/2023/05/22/1177590231/fake-viral-images-of-an-explosion-at-the-pentagon-were-probably-created-by-ai>, 2023. 1
- [6] Geoff Brumfiel. How AI-generated content increased disinformation after Maduro’s removal. <https://www.npr.org/2026/01/10/nx-s1-5669478/how-ai-generated-content-increased-disinformation-after-maduros-removal>, 2026. 8
- [7] Hany Farid. Mitigating the harms of manipulated media: Confronting deepfakes and digital deception. *PNAS Nexus*, 4(7):pgaf194, 2025. 1
- [8] Gunnar Farneback. Two-frame motion estimation based on polynomial expansion. In *Image Analysis*, pages 363–370, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg. 5
- [9] Lonsdale Augustus Hale. *The fog of war*. 1896. 8
- [10] Olivier J Hénaff, Robbe LT Goris, and Eero P Simoncelli. Perceptual straightening of natural videos. *Nature Neuroscience*, 22(6):984–991, 2019. 3
- [11] Tiffany Hsu and Stuart A Thompson. AI muddies Israel-Hamas war in unexpected way. <https://www.nytimes.com/2023/10/28/business/media/ai-muddies-israel-hamas-war-in-unexpected-way.html>, 2023. 1
- [12] Christian Internö, Robert Geirhos, Markus Olhofer, Sunny Liu, Barbara Hammer, and David Klindt. AI-generated video detection via perceptual straightening. arXiv:2507.00583, 2025. 3
- [13] S Rao Jammalamadaka and Ambar Sengupta. *Topics in circular statistics*. World Scientific, 2001. 6
- [14] Trapeznikov Kirill, Paul Cummer, Jonathan Li, Gabriel Mancino-Ball, Jai Aslam, Michael Davinroy, Peter Bautista, Laura Cassani, Danial Vahdati, Tai Nguyen, Matthew Stamm, Jill Crisman, and Jesse Hostetler. SAFE: Video challenge. <https://safe-video-2025.dsri.org>, 2025. 1
- [15] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 4
- [16] David Klepper. Fake babies, real horror: Deepfakes from the Gaza war increase fears about AI’s power to mislead. <https://apnews.com/article/artificial-intelligence-hamas-israel-misinformation-ai-gaza-a1bb303b637ffbbb9cbc3aa1e000db47>, 2023. 1
- [17] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55. Springer, 2024. 4
- [18] Liangcun Jiang Dayu Yu Tianyu Tuo Jian Li Ming Wang, Peng Yue. FASDD: An open-access 100,000-level flame and smoke detection dataset for deep learning in fire detection. *Scientific Data*, 2024. 2
- [19] Huy H Nguyen, Junichi Yamagishi, and Isao Echizen. Exploring self-supervised vision transformers for deepfake detection: A comparative analysis. In *IEEE International Joint Conference on Biometrics*, pages 1–10. IEEE, 2024. 3
- [20] Sophie J Nightingale and Hany Farid. AI-synthesized faces are indistinguishable from real faces and more trustworthy. *Proceedings of the National Academy of Sciences*, 119(8):e2120481119, 2022. 1
- [21] Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. Towards universal fake image detectors that generalize across generative models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24480–24489, 2023. 3
- [22] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. arXiv:2304.07193, 2023. 3
- [23] Ed Pizzi, Giorgos Kordopatis-Zilos, Hiral Patel, Gheorghe Postelnicu, Sugosh Nagavara Ravindra, Akshay Gupta, Symeon Papadopoulos, Giorgos Tolias, and Matthijs Douze. The 2023 video similarity dataset and challenge. *Computer Vision and Image Understanding*, 243:103997, 2024. 2
- [24] Runway. The Turing reel. <https://runwayml.com/research/theturingreel>, 2025. Accessed February 2026. 2
- [25] Schey. Div, grad, curl, and all that: An informal text on vector calculus, 2005. 5
- [26] Eero P Simoncelli. Bayesian multi-scale differential optical flow. *Handbook of Computer Vision and Applications: Signal processing and pattern recognition*, 2:397, 1999. 4
- [27] Pyronear Team. Pyro-SDIS dataset. <https://huggingface.co/pyronear/pyro-sdis>, 2024. 2
- [28] Wenhao Wang and Yi Yang. Vidprom: A million-scale real prompt-gallery dataset for text-to-video diffusion models. *Advances in Neural Information Processing Systems*, 37:65618–65642, 2024. 2