Automated Visual Change Detection for Film Continuity Analysis using Training Data Derived from Composite Error Examples

Asmita Manna[1], Swarup Kusalkar[2], Vallabh Khomane[3], Kedar Kolambe[4], Apurv Kawale[5]
[1,2,3,4,5]*Department of Computer Engineering Pimpri Chinchwad College of Engineering (PCCOE) Pune, Maharashtra, India*

***Abstract*—Visual continuity errors are prevalent inconsisten- cies in film and video production that require meticulous manual checking by dedicated on-set and post-production personnel. This paper presents a deep learning approach to automate the detection of visual changes that may correspond to such errors. Motivated by the availability of composite images illus- trating known continuity mistakes—typically showing side-by- side frames with superimposed highlights—we propose a system that bypasses the need for manual pixel-level annotation. An automated preprocessing pipeline extracts corresponding image patches guided by visual highlights in the source composites and generates binary change masks based on thresholded pixel differences between the patches. A U-Net-like architecture, fea- turing shared encoder weights and difference-based skip con- nections utilising a custom AbsoluteValueLayer for robust serialisation, is trained on these automatically derived data triplets. The trained model produces a probability map indicating pixel-wise changes between two new input images. We detail the automated data-generation strategy, the network design, and evaluation results using standard segmentation metrics. We discuss the system's effectiveness in localising significant pixel discrepancies and acknowledge the inherent limitations arising from the automated mask-generation process, positioning the tool as an aid for accelerating manual continuity review.**

***Index Terms*—Change Detection, Continuity Error, Deep Learning, U-Net, Siamese Network, Computer Vision, Video Analysis, Automated Preprocessing, Image Segmentation.**

I. INTRODUCTION

The seamless flow of visual information is fundamental to narrative coherence in film and television. Disruptions in this flow—known as *continuity errors*—arise when elements within a scene change inconsistently between shots: props may vanish, reappear, or change state; costumes may be altered; or the positions of actors and objects may inexplicably shift [1]. While often subtle, such errors can break viewer immer- sion and diminish perceived production quality. Identifying and preventing them traditionally falls to script supervisors during filming and to editors in post-production, demanding painstaking manual comparison and record-keeping [2]. This process is inherently time-consuming, costly, and susceptible to oversight, particularly given the scale of modern digital productions [3].

Automating the detection of visual inconsistencies offers a path towards enhancing the efficiency and reliability of conti- nuity verification. Change detection—the task of identifying differences between images of the same scene acquired at different times or viewpoints [4]—provides a natural compu- tational framework for this problem. With the advent of deep convolutional networks, image-level and pixel-level change localisation has advanced substantially [5], [6].

Prior work by Pickup and Zisserman [7] demonstrated auto- matic retrieval of continuity errors by analysing shot structure, registering candidate shot pairs via homographies, suppressing changes due to detected humans, and examining residual pixel differences. Their approach relies on robust registration and explicit filtering of intentional motion.

Our work leverages a different type of source data: com- posite images readily available online [8] that highlight known errors by juxtaposing *before* and

after frames with coloured markers. While such images identify errors, they are unsuitable for directly training standard detection models, and manually creating precise pixel-level change masks for a large dataset remains prohibitive.

We propose a system that automatically generates training data from these composite examples. Our key contributions are:

1) An automated preprocessing pipeline that extracts cor- responding image patches guided by visual highlights (e.g., coloured circles) present in the composite source images.
2) A method for automatically generating binary change masks by thresholding the pixel-wise absolute difference between the extracted *before* and *after* patches, thereby eliminating manual annotation.
3) A U-Net-based change detection network with shared encoder weights and difference-comparison skip connections implemented via a custom AbsoluteValueLayer for reliable model serialisation.
4) Demonstration of the trained model's ability to localise significant pixel discrepancies between new input image pairs, serving as an automated flag for human review.

We acknowledge that training on automatically generated difference masks produces a model that detects *any* sufficiently large pixel discrepancy rather than specific semantic errors. The paper details the methodology, data generation, network architecture, and experimental validation.

II. RELATED WORK

Detecting continuity errors intersects with image change detection, video analysis, semantic segmentation, and human detection.

A. Image Change Detection

Change detection is a well-studied problem in remote sensing and video surveillance [4]. Classical methods in- clude pixel-wise differencing [9] and statistical change vec- tor analysis [10]. Deep learning approaches—in particular Siamese networks [11], [12] and U-Net variants [13], [14]— have achieved superior performance by learning discriminative representations jointly from image pairs. Recent transformer- based architectures further improve the modelling of long- range context [15], [16].

B. Semantic Segmentation and Scene Understanding

Semantic segmentation underpins many change detection pipelines by providing object-level priors [17], [18]. Dilated convolutions [19] capture multi-scale context without sacri- ficing spatial resolution—a principle reflected in our encoder design. Non-local attention mechanisms [20] further help models focus on semantically relevant regions.

C. Continuity Error Analysis in Film

Pickup and Zisserman [7] provided the foundational ap- proach: they exploited ABAB shot structure [2], computed homographies via RANSAC [21], suppressed human-occupied regions using HOG-based detection [22], [23], and examined residual differences. Related work aligns movie scripts with video content [24] or applies segmentation to verify object hypotheses [25]. Human detection methods—from Viola– Jones [26] to modern deep detectors [27]—remain relevant for isolating intentional from unintentional motion.

Our approach differs by starting from pre-identified error composites, using automated difference masking for super- vision, and employing a modern end-to-end deep learning architecture rather than hand-crafted registration and filtering pipelines.

D. Self-Supervised and Contrastive Representation Learning

Self-supervised methods [28], [29] learn visual similarity without labels—a direction relevant to future extensions of this work where paired training data may be scarce.

III. METHODOLOGY

A. System Overview

The system operates in two phases. An offline prepro- cessing step automatically converts source composite images into (patch_a, patch_b, change_mask) triplets. Subse- quently, a U-Net model is trained on these triplets. During inference, the trained model accepts two new images, prepro-cesses them, predicts a change mask, and annotates the result with detected regions of change.

B. Automated Dataset Preprocessing

This step translates annotated composite images into train- able data without manual mask drawing.

1) Input: Composite images from the source directory.
2) Frame Splitting: Each image is divided into an assumed top (Frame A) and bottom (Frame B) half.
3) Highlight Detection: Contours matching a specified HSV range (e.g., purple) and a minimum area threshold are located; centroids are computed.
4) Highlight Matching: Highlights in A and B are paired by centroid proximity constrained by a configurable distance factor.
5) Patch Extraction: Square patches (e.g., 256×256) centered on matched highlight centroids are extracted from Frame A (patch_a) and Frame B (patch_b).
6) Automated Mask Generation: cv2.absdiff(patch_a, patch_b) is converted to grayscale and binarised via a fixed threshold or Otsu's method [30] to yield change_mask.png.
7) Output: The triplet is written to the train/ or validation/subdirectory.

C. Change Detection Network Architecture

The proposed network follows a U-Net-based design adapted for change detection between image pairs.

- Inputs: patch_a and patch_b, both of shape $H \times W \times 3$.
- Shared Encoder: Four encoder blocks with shared weights process both inputs. Each block contains two Conv2D layers (Batch Normalisation, ReLU) followed by MaxPooling2D; pre-pooling feature maps are retained for skip connections.
- Difference Computation: At the bottleneck and at each skip level, features from the two branches are subtracted via layers.Subtract, and the absolute value is taken via the custom AbsoluteValueLayer.
- Decoder: Four decoder blocks progressively upsample

(Conv2DTranspose) features from the bottleneck difference. Each stage concatenates the upsampled map with the corresponding absolute-difference skip map, followed by a convolutional block.

- Output: A 1×1 Conv2D with sigmoid activation produces a single-channel probability map at the inputresolution.

The AbsoluteValueLayer (registered as a Keras custom object) replaces a standard Lambda layer, ensuring reli- able saving and loading in the .keras format.

D. Loss Function

We employ Dice Loss [31] as the optimisation objective:

$$L_{\text{Dice}} = 1 - \frac{2 \sum_i p_i g_i + \varepsilon}{\sum_i p_i + \sum_i g_i + \varepsilon}, \quad (1)$$

where p_i and g_i denote the predicted probability and ground- truth label at pixel i, and ε is a smoothing constant. Dice loss is well-suited to the class imbalance typical of change detection, where changed pixels are sparse [32].

E. Training Procedure

- Data Pipeline: Triplets are loaded via tf. data. Datasetfor efficient streaming.
- Optimiser: Adam [33] with learning rate 10^{-4}.

Callbacks: Model Checkpoint saves the best weights according to validation Dice coefficient; Early Stoppingwith patience 10 prevents overfitting; Tensor Board logs are recorded.

F. Inference and Annotation

- Model Loading: The. keras model is loaded with the CUSTOM_OBJECTS dictionary containing AbsoluteValueLayer, dice_loss, and dice_coeff.
- Preprocessing: Input images are resized to the model's input shape.
- Alignment (Optional): ORB-based homography estima- tion can align image B to image A (experimental).
- Prediction: The model outputs a probability mask, thresholded at 0.5.
- Annotation: Contours of the change region are drawn on image A (or an overlay is applied) to flag areas for review.

IV. DATASET DESCRIPTION

The primary source data consist of approximately 700 composite images collected from publicly available online repositories documenting film errors

(e.g., MovieMistakes.com [8]) and manual collection from specific productions. Each composite presents two frames (top/bottom or left/right), il- lustrates a visual continuity error, and includes superimposed coloured circles or boxes highlighting the region of interest along with a short textual description. Error types include ob- ject disappearances and appearances, colour shifts, positional jumps, and minor prop inconsistencies.

The automated pipeline (Section III-B) processed this source data to yield a dataset of image patch triplets:

- Training Set: approximately 354 triplets.
- Validation Set: approximately 88 triplets.

Each triplet contains patch_a.png and patch_b.png (both 256×256 px) and the automatically generated change_mask.png. Critically, change_mask.png reflects pixel-difference magnitude rather than semantic error boundaries; this distinction is central to interpreting the evaluation results.

V. EXPERIMENTS AND RESULTS

A. Experimental Setup

- Hardware: Intel Core i7 CPU, 16 GB RAM, NVIDIA GeForce GTX 1660 Ti GPU (6 GB VRAM).
- Software: Python 3.10, TensorFlow 2.15.0, Keras 3.0.2,

OpenCV 4.9.0, NumPy 1.26.

- Training: Input shape 256×256×3, base filters 32, batch size 8, Adam (lr = $1\,0^{-4}$), Dice Loss, up to 50 epochs with early stopping (patience 10) on val_dice_coeff.
- Data: 354 training / 88 validation triplets.

B. Evaluation Metrics

Performance was assessed on the validation set using:

- Dice Coefficient: Dice = $\frac{2|P \cap G|}{|\hat{P}| + |G|}$ [34], the primary overlap metric.
- Intersection over Union (IoU): IoU = $\frac{|\hat{P} \cap G|}{|\hat{P} \cup G|}$ [35], a stricter overlap measure.

Binary masks were obtained by thresholding predicted proba- bility maps at 0.5.

C. Quantitative Results

Table I reports mean validation performance.

Metric	Mean Score
Dice Coefficient	0.72
IoU (Jaccard)	0.61

Evaluated against automatically generated masks. Threshold = 0.5.

A Dice score of 0.72 indicates that the model successfully learned to reproduce the automatically generated difference masks. The lower IoU (0.61) is expected given the stricter intersection criterion and suggests modest over-prediction at change boundaries.

D. Qualitative Results

Inference results on held-out image pairs demonstrate three characteristic behaviours. In successful cases, the model ac- curately localises the changed object within the scene. In cases involving minor lighting differences, the model exhibits sensitivity to such variations within the original highlight area. When alignment between input frames is disabled, residual misalignment can dominate the detected region, underscoring the importance of geometric registration as a preprocessing step.

VI. DISCUSSION

The results demonstrate the technical feasibility of training a change detection network on data automatically derived from composite error examples. The model achieves Dice/IoU scores (Table I) indicating successful learning of the patterns encoded in the auto-generated masks, and visually localises regions of noticeable pixel change between input pairs.

The core strength of this methodology is the significant re- duction in manual effort. By automatically generating training masks, we bypass the laborious pixel-level annotation required for traditional supervised approaches [11]. This enables ex- ploiting readily available composite error illustrations at scale. The primary limitation stems directly from this automation.

A model trained on difference masks learns to detect *any* sufficiently large pixel discrepancy, irrespective of semantic cause. It does not inherently distinguish a genuine continuity error from intentional character

motion, lighting fluctuations, compression artefacts, or background shifts. The model there- fore functions as a general-purpose *difference localiser* rather than a semantic *continuity error detector*.

Accordingly, Dice and IoU metrics measure fidelity to the potentially noisy difference mask, not accuracy in isolating se- mantically meaningful errors. Incorporating higher-level scene understanding—e.g., via pretrained object detectors [27] or vision-language models—could improve semantic precision in future iterations.

Image alignment during inference remains critical. Without robust registration, geometric shifts between frames dominate the detected differences. Learned alignment modules [36] represent a promising extension.

Despite these limitations, the system holds practical value as an assistive tool: by automatically flagging regions of significant visual change between related frames, it can sub- stantially accelerate manual review for continuity supervisors and editors.

VII. CONCLUSION AND FUTURE WORK

We presented an end-to-end deep learning system for de- tecting visual changes between image pairs, targeting the automated identification of film continuity errors. A key con- tribution is an automated preprocessing pipeline that generates training data—image patches and pixel-difference masks— from composite error illustrations, entirely avoiding manual annotation. A U-Net based network trained on this data successfully localises regions of significant pixel discrepancy, achieving a mean Dice coefficient of 0.72 on the validation set.

While the automated data generation enables rapid dataset construction, it produces a model that detects general pixel differences rather than specific semantic errors. The practical value lies in its potential to accelerate manual continuity checks by highlighting areas of potential inconsistency for human review. Future research directions include:

- Refined Automated Masking: Incorporating unsuper- vised segmentation or OCR-derived textual cues to gen- erate semantically targeted change masks.
- Human-in-the-Loop Refinement: Lightweight annota- tion tools enabling users to accept, reject, or refine masks iteratively to build a higher-quality dataset.
- Robust Alignment: Evaluating learned image registra- tion modules [36] for misaligned inference inputs.
- Temporal Modelling: Extending the architecture to anal- yse short frame sequences to better differentiate transient motion from persistent changes [37].
- Transformer Architectures: Investigating transformer- based change detection models [15], [16] that capture long-range spatial dependencies.
- Semantic Supervision: Integrating pretrained object de- tectors [27] to distinguish continuity errors from inten- tional scene changes.

This work provides a foundation for further exploration of automated and semi-automated techniques for managing visual continuity in media production.

REFERENCES

[1] T. Smith, An attentional theory of continuity editing. University of Edinburgh, 2006, phD dissertation.

[2] J. Monaco, How to Read a Film: Movies, Media, Multimedia, 3rd ed. New York, NY, USA: Oxford University Press, 2000.

[3] D. Bordwell, K. Thompson, and J. Smith, Film Art: An Introduction, 12th ed. New York, NY, USA: McGraw-Hill Education, 2019.

[4] M. Hussain, D. Chen, A. Cheng, H. Wei, and D. Stanley, "Change detection from remotely sensed images: From pixel-based to object-based approaches," in ISPRS Journal of Photogrammetry and Remote Sensing, vol. 80. Elsevier, 2013, pp. 91–106.

[5] H. Chen and Z. Shi, "A spatial-temporal attention-based method and a new dataset for remote sensing image change detection," Remote Sensing, vol. 12, no. 10, p. 1662, 2020.

[6] Q. Shi, M. Liu, S. Li, X. Liu, F. Wang, and L. Zhang, "A deeply supervised attention metric-based network and an open aerial image dataset for remote sensing change detection," IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1–16, 2022.

[7] L. C. Pickup and A. Zisserman, "Automatic retrieval of visual continuity errors in movies," in Proc. ACM International Conference on

Image and Video Retrieval (CIVR). ACM, 2009, pp. 1–8.

[8] MovieMistakes.com, "Movie mistakes — continuity errors," https:// www.moviemistakes.com, 2024, accessed: 2024.

[9] P. L. Rosin and G. A. W. West, "Thresholding for change detection," Computer Vision and Image Understanding, vol. 86, no. 2, pp. 79–95, 2002.

[10] W. A. Malila, "Change vector analysis: An approach for detecting forest changes with Landsat," in Proc. LARS Symposia, 1980, pp. 385–397.

[11] R. C. Daudt, B. L. Saux, and A. Boulch, "Fully convolutional Siamese networks for change detection," in Proc. IEEE International Conference on Image Processing (ICIP). IEEE, 2018, pp. 4063–4067.

[12] Y. Zhan, K. Fu, M. Yan, X. Sun, H. Wang, and X. Qiu, "Change detection based on deep siamese convolutional network for optical aerial images," IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 10, pp. 1845–1849, 2017.

[13] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional net- works for biomedical image segmentation," in Proc. International Con- ference on Medical Image Computing and Computer-Assisted Interven- tion (MICCAI). Springer, 2015, pp. 234–241.

[14] S. Ji, S. Wei, and M. Lu, "Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set," IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 1, pp. 574–586, 2019.

[15] H. Chen, Z. Qi, and Z. Shi, "Remote sensing image change detection with transformers," in IEEE Transactions on Geoscience and Remote Sensing, vol. 60. IEEE, 2021, pp. 1–14.

[16] W. G. C. Bandara and V. M. Patel, "A transformer-based Siamese network for change detection," arXiv preprint arXiv:2201.01293, 2022.

[17] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2015, pp. 3431–3440.

[18] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2017, pp. 2881–2890.

[19] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 4, pp. 834–848, 2018.

[20] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local means net- works," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2018, pp. 7794–7803.

[21] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, 2nd ed. Cambridge, UK: Cambridge University Press, 2004.

[22] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1. IEEE, 2005, pp. 886–893.

[23] V. Ferrari, M. Marin-Jimenez, and A. Zisserman, "Progressive search space reduction for human pose estimation," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2008, pp. 1–8.

[24] T. Cour, C. Jordan, E. Miltsakaki, and B. Taskar, "Movie/script: Align- ment and parsing of video and text transcription," in Proc. European Conference on Computer Vision (ECCV). Springer, 2008, pp. 158– 171.

[25] D. Ramanan, "Using segmentation to verify object hypotheses," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2007, pp. 1–8.

[26] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1. IEEE, 2001, pp. 511–518.

[27] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real- time object detection with region proposal networks," in Advances in Neural Information Processing Systems

(NeurIPS), vol. 28. Curran Associates, 2015, pp. 91–99.

[28] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in Proc. International Conference on Machine Learning (ICML). PMLR, 2020, pp. 1597– 1607.

[29] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2020, pp. 9729–9738.

[30] N. Otsu, "A threshold selection method from gray-level histograms," IEEE Transactions on Systems, Man, and Cybernetics, vol. 9, no. 1, pp. 62–66, 1979.

[31] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," in Proc. International Conference on 3D Vision (3DV). IEEE, 2016, pp. 565– 571.

[32] S. Jadon, "A survey of loss functions for semantic segmentation," arXiv preprint arXiv:2006.14822, 2020.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in Proc. International Conference on Learning Representations (ICLR), 2015.

[34] L. R. Dice, "Measures of the amount of ecologic association between species," Ecology, vol. 26, no. 3, pp. 297–302, 1945.

[35] P. Jaccard, "The distribution of flora in the alpine zone," New Phytolo- gist, vol. 11, no. 2, pp. 37–50, 1912.

[36] I. Rocco, R. Arandjelovic, and J. Sivic, "Convolutional neural network architecture for geometric matching," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2017, pp. 6148–6157.

[37] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venu- gopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in Proc. IEEE Con- ference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2015, pp. 2625–2634.