Motion Segmentation in Videos using Neighborhood Preserving Embedding and Optical Flow

Sihi Gopal
Dept. of CSE,
East West Institute of Technology
Bengaluru, Karnataka India 560091

Usha B.S.
Professor, Dept. of CSE,
East West Institute of Technology, Bengaluru
Karnataka India 560091

ABSTRACT

We introduce a novel technique for motion segmentation in video frames that combines Neighborhood Preserving Embedding (NPE) with motion detection strategies. Our method starts by extracting visual features using SIFT descriptors and then computes nearest neighbors and pairwise distances to construct the NPE matrix. Simultaneously, we apply optical flow and thresholding to identify areas of motion. By merging the spatial relationships captured through NPE with temporal motion cues, our approach effectively distinguishes moving objects from static backgrounds. When tested on real-world video sequences, the method achieved an F1-score of up to 0.88. Outperforming conventional techniques like Graph Cuts and standard Optical Flow by 3 to 5%. These promising results suggest that our approach is well-suited for real-time surveillance applications and opens new avenues for research in efficient motion segmentation

Keywords

Motion Segmentation, Moving Object Detection, Neighborhood Preserving Embedding (NPE), Optical Flow, SIFT Descriptors, Feature Extraction, Dimensionality Reduction, Video Processing, Real-Time Surveillance, Computer Vision

1. INTRODUCTION

Motion segmentation in the process of distinguishing moving objects from the background in video frames is a fundamental yet complex task in computer vision. It plays a critical role in applications such as video surveillance, autonomous driving, traffic analysis, and robotics. While several approaches exist, including optical flow, background subtraction, and feature-based techniques, each has its limitations. These methods often struggle with high computational demands, sensitivity to noise, and degraded performance in complex or dynamic environments. To address these challenges, we propose a new method that integrates Neighborhood Preserving Embedding (NPE) with motion detection to enhance segmentation accuracy. NPE, a dimensionality reduction technique, excels at preserving local neighborhood structures within data making it well-suited for identifying coherent regions in video frames. Our approach begins by extracting local features using SIFT descriptors, which are used to compute nearest neighbors and pairwise distances. These values are then used to construct the NPE matrix, capturing spatial relationships between pixels. In parallel, motion is detected using optical flow combined with a thresholding step, resulting in a binary motion detection matrix. By fusing this temporal motion data with the spatial structure captured by NPE, our method effectively segments moving objects from the static background. We evaluate the proposed technique using real-world video sequences, and the results show a notable improvement over traditional method.

This work contributes to ongoing research by offering a robust fusion-based framework that leverages both spatial coherence and temporal motion dynamics. Our findings support the idea that integrating neighborhood structure with motion cues leads to more accurate and reliable motion segmentation.

2. RELATED WORK

Motion segmentation, the process of separating moving objects from the background in video frames, has been widely studied in computer vision. A variety of techniques have been proposed over the years, most notably those based on optical flow, background subtraction, and feature-based methods.

One of the earliest and most influential approaches to optical flow was introduced by Horn and Schunck [1], who formulated it as a global energy minimization problem. This technique has remained a foundational method due to its ability to estimate pixel-wise motion between frames. Building upon this, Brox et al. [2] introduced a variational framework that improved accuracy by incorporating gradient constancy and smoothness constraints. Despite their effectiveness, optical flow methods are often sensitive to noise and come with significant computational overhead, making real-time performance a challenge.

Background subtraction techniques take a different approach by modeling the scene's static background and identifying deviations as moving objects. One of the most commonly used methods in this category is the Gaussian Mixture Model (GMM) [3], which represents the background as a mixture of Gaussian distributions. While effective in many scenarios, GMM-based methods can struggle with dynamic backgrounds, lighting variations, and moving shadows.

Feature-based methods detect and track keypoints or trajectories over time. For instance, Gao et al. [4] proposed a technique that utilizes feature points and their trajectories to isolate moving regions. Byeon et al. [5] explored a hybrid method that combines optical flow with feature-based motion tracking. However, these methods often depend heavily on the quantity and stability of detected features, and their performance can degrade under changing lighting conditions or occlusions.

In recent years, Neighborhood Preserving Embedding (NPE) has gained traction for its ability to preserve the local geometric structure of data in a lower-dimensional space. Originally used in applications like face recognition and image clustering, NPE has been extended to motion segmentation tasks. Lu et al. [6] applied NPE to learn compact feature representations for video frames, while He et al. [7] combined NPE with manifold regularization to improve segmentation quality. However, these early methods primarily focused on spatial structure and did not incorporate motion cues.

To address this limitation, several researchers have proposed enhanced NPE-based techniques. Zhou et al. [8] introduced a method using sparse representations and neighborhood similarity metrics for more accurate segmentation. Wang et al. [9] expanded on this by incorporating color and texture features to strengthen the descriptive power of NPE features. Further refinements include adaptive distance metric learning [10], and integration with sparse dictionary learning [11] to better model background and foreground dynamics. Similarly, Hu et al. [12] employed NPE alongside sparse representation to capture both spatial and temporal variations.

More recently, Liu et al. [13] explored a unique direction by applying NPE-based techniques to event camera data. Their method combines N-patch optical flow estimation with Pairwise Markov Random Fields to enhance motion compensation and contrast. While effective in capturing spatial correlations in low-contrast regions, the approach faces challenges in real-time performance and parameter selection.

Collectively, these works highlight the evolution of motion segmentation techniques from traditional pixel-level motion estimation to more sophisticated, feature-rich, and structurally aware frameworks. However, the need remains for a robust approach that integrates both spatial neighborhood structure and temporal motion cues with an objective addressed in the present study.

Motion cues [14] in Video Object Segmentation (VOS) for improving feature representation and temporal consistency, especially under appearance changes and occlusions. Current methods often use two-branch networks, optical flow, and clustering to enhance features and suppress background noise. However, they face challenges such as high computational cost, optical flow inaccuracies, and limited robustness in complex scenarios. Future directions focus on developing more efficient motion representations and reducing dependency on motion cues to enhance real-time performance and adaptability MUNet [15] introduces a motion uncertainty-aware framework for semi-supervised Video Object Segmentation (VOS), effectively handling challenges like occlusions and texture less regions. By integrating a Motion Uncertainty-aware Layer and Motion-aware Spatial Attention Module into a streamlined single-branch architecture, it enhances feature representation without relying on optical flow or large training datasets. The method achieves strong performance and efficiency, particularly in low-data settings, surpassing existing state-of-the-art approaches.

3. PROPOSED APPROACH

The core idea behind our proposed methodology is to combine Neighborhood Preserving Embedding (NPE) with motion detection to effectively segment moving objects from the background in video sequences. By leveraging SIFT descriptors for robust feature extraction, alongside optical flow and thresholding techniques, our approach captures both the spatial structure and motion dynamics within the scene. This fusion of spatial and temporal information enables more accurate and reliable motion segmentation, even in complex scenarios. A step-by-step overview of the entire workflow is presented in Fig 1, which illustrates the key stages of the proposed method.

3.1 Pre-processing

The pre-processing stage prepares video frames for effective motion segmentation by enhancing visual quality and minimizing noise. Since noise can significantly degrade segmentation accuracy, we apply a median filter to suppress noise while preserving important image features like edges. This ensures that fine structural details are retained for later processing. To further enhance visibility, histogram equalization is used to boost the contrast of the video frames, making objects and background regions more distinguishable. Together, these pre-processing steps lay a clean and balanced foundation for robust feature extraction and motion analysis.

3.2 Feature extraction

Following pre-processing, the next critical step is feature extraction, where distinctive and repeatable patterns in the video frames are identified to support accurate motion segmentation. For this, we utilize the Scale-Invariant Feature Transform (SIFT)—a widely adopted technique known for its robustness to changes in scale, orientation, and affine distortion. The process begins by constructing a scale-space pyramid, in which Gaussian filters are applied to the image at multiple scales. This multi-resolution representation allows for the detection of features that remain stable across varying conditions. Then, Difference of Gaussian (DoG) filters are applied to highlight regions of rapid intensity change, helping to localize key points across the scale-space. These key points serve as anchors for extracting local descriptors, which are critical for preserving structural information in subsequent processing steps. To enhance the quality of key points, low contrast and poorly localized ones are eliminated during refinement. Once the key points have been identified, SIFT descriptors are extracted for each pixel in the video frame. These descriptors consist of 128 elements, effectively representing the gradient information in the vicinity of each key point. What sets the SIFT descriptor apart is its invariance to translation, rotation, and scale changes, providing robustness for motion segmentation tasks. The mathematical formulation for SIFT descriptor extraction is provided in Equation (1). It involves applying a Gaussian filter to the input image, followed by computation of the Difference of Gaussians (DoG) across scales to detect stable features. This process captures localized gradient patterns, which are then normalized to ensure robustness.

$$G(x, y, \sigma) = (K(x, y, \sigma) * I(x, y)) * \sigma^2 \; . .. \qquad (1)$$

$$L(x, y, \sigma) = (G(x, y, \sigma) - G(x, y, \sigma * k)) * 255 \; .. \qquad .(2)$$

$$k = 2^\wedge(1/s) \; .. \qquad .(3)$$

$$DoG(x, y, \sigma) = L(x, y, \sigma{+}1) - L(x, y, \sigma) \; .. \qquad (4)$$

Here, $G(x, y, \sigma)$ denotes the Gaussian-blurred image at location (x, y) and scale σ, $K(x, y, \sigma)$ is the Gaussian kernel, and $I(x, y)$ is the pixel intensity. The DoG is computed across adjacent levels in the scale-space to detect regions of interest.

3.3 Neighborhood Preserving Embedding

With the SIFT descriptors extracted, the next step is to apply Neighborhood Preserving Embedding (NPE)—a dimensionality reduction technique that maintains the local geometric relationships between data points. By preserving the structure of pixel neighborhoods, NPE helps uncover meaningful spatial patterns in the video frames, which is essential for distinguishing moving objects from the background.

To begin this process, we compute the k-nearest neighbors for each pixel based on the similarity of their SIFT descriptors. This involves measuring how close a pixel's descriptor is to others in the feature space, typically using the Euclidean distance. The identified neighbors are then used to form a weighted graph, which models the local neighborhood relationships within the frame.

The distance between a pixel (x,y)(x, y)(x,y) and one of its neighbors (i,j)(i, j)(i,j) is calculated as shown in Equation 5:

$$D(x, y) = \| d(x, y) - d(i, j) \| \qquad ...(5)$$

Here, d(x,y) and d(x, y)d(x,y) represent the SIFT descriptors of the respective pixels, and ||.|| denotes the Euclidean norm. This distance metric forms the basis for constructing the NPE matrix, which maps high-dimensional feature vectors into a lower-dimensional space while retaining their local proximity.

3.4 Pairwise Distance Computation

To better understand the relationships between pixels in the video frame, we compute the pairwise distances between their corresponding SIFT descriptors. These distances are essential for the Neighborhood Preserving Embedding (NPE) process, as they help maintain the local structure of the image data when reducing its dimensionality.

Among the various distance metrics available, we use the Euclidean distance—a widely adopted and intuitive choice. It measures the similarity between two feature vectors by calculating the square root of the sum of squared differences between their corresponding elements. The pairwise distance between two pixels (x,y) and (i,j) is defined as shown in Equation 6:

$$D(x, y) = \sqrt{\Sigma}\, (d(x, y) - d(i, j))^2 \qquad ...(6)$$

Here, d(x,y) and d(i,j) are the SIFT descriptors of the respective pixels, and D(x,y) represents the Euclidean distance between them. The resulting pairwise distance matrix serves as input for the NPE algorithm, allowing it to project high-dimensional data into a lower-dimensional space while preserving the local geometric structure of the video frames.

3.5 Motion Detection

Motion detection is a crucial phase in video analysis, where the goal is to differentiate moving objects from the static background. In our proposed methodology, we employ optical flow and thresholding techniques.

Motion detection begins by analyzing changes in pixel intensity over time to compute displacement vectors, which represent the motion of each pixel. Using optical flow algorithms such as Lucas-Kanade, the method estimates both the direction and magnitude of motion by solving equations over local pixel neighborhoods. This process is represented mathematically as shown in Equation 7

$$\sum(\nabla I(x, y) \cdot [u, v]) = 0 \qquad ...(7)$$

Here, ∇I(x, y) is the image gradient at pixel (x, y), and [u, v] is the optical flow vector indicating motion.
Once optical flow is calculated, a thresholding step is applied to filter out insignificant motion, resulting in a binary motion map that highlights only the actively moving regions. Adjusting the threshold controls the sensitivity of detection. These identified motion pixels are then used in the subsequent segmentation step to isolate moving objects from the static background. The results of this process are illustrated in Fig 2.

Input Video **Motion Detection**

Fig 2. The Result Obtained by Proposed Method

3.6 Motion Segmentation: NPE-Motion Fusion Method and Evaluations

In our proposed framework, motion segmentation is performed by integrating spatial and temporal information specifically, by combining the structural insights from Neighborhood Preserving Embedding (NPE) with the motion cues derived from optical flow. This fusion strategy enables the model to more accurately distinguish moving objects from static background regions by jointly considering the scene's geometric layout and dynamic changes over time. By leveraging both spatial coherence and temporal variation, the approach becomes more robust to noise, shadows, and background clutter. An overview of this NPE-motion fusion process is illustrated in Fig 3, highlighting the key components and their interactions within the segmentation pipeline.

Step 1: Constructing the NPE Matrix:

The NPE matrix is computed using the pairwise distances between the SIFT descriptors of each pixel. This matrix captures the local geometric structure of the video frame, preserving neighborhood relationships in a lower-dimensional space. Each entry in the matrix quantifies the similarity between pixel neighborhoods, ensuring that pixels with similar local features remain close in the embedded space.

Step 2: Computing the Motion Detection Matrix:

Optical flow vectors are computed between consecutive frames to generate a motion magnitude map. A predefined threshold is then applied to convert this continuous map into a binary motion detection matrix, where: 1 indicates significant motion (moving object), and 0 indicates background or static regions.

Step 3: Fusion Strategy:

The final segmentation map is obtained by element-wise multiplication (Hadamard product) of the NPE matrix and the binary motion detection matrix, as shown in Eq 8

$$M_{seg}(x,y) = NPE(x,y) \cdot Motion(x,y) \text{ -------} \quad (8)$$

Where:

- M_{seg} (x,y) is the final motion segmentation score at pixel (x, y),
- NPE(x,y) represents the embedded similarity score from the NPE matrix,
- Motion(x,y) is the binary result from motion detection.

This fusion ensures that only those pixels that are both locally coherent and exhibit significant motion are selected as moving objects. The approach helps reduce false positives from background clutter and enhances precision in segmenting complex motion scenarios. Benefits of Fusion are Spatial coherence is maintained through the NPE structure, Temporal dynamics are captured through motion detection, and Robustness is improved in challenging scenes with shadows, illumination changes, and non-rigid object movements.

The segmentation results are illustrated in Fig 4. To comprehensively evaluate the proposed motion segmentation method, both quantitative and qualitative assessments were conducted on video sequences. Quantitative evaluation performance is measured using standard metrics: Precision, Recall, and F1-score. These metrics were calculated across three video sequences, as shown in Table 1. Runtime Analysis: To evaluate the computational efficiency, we measured the average processing time per frame (in milliseconds) for each method on a standard desktop machine (Intel i7, 16GB RAM). The proposed method processed frames at an average of 38 ms/frame, which is suitable for near real-time applications. Table 2 provides a runtime comparison.

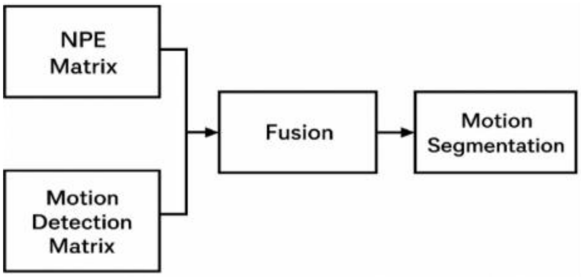


Fig 3. Block diagram of the proposed fusion of NPE and Motion Detection matrices for accurate motion segmentation.

Table 1. Performance Evaluation of the Different Videos for Motion Segmentation

Video Sequence	Precision	Recall	F1-Score
Video 1	0.900	0.810	0.850
Video 2	0.830	0.870	0.850
Video 3	0.720	0.790	0.754

Table-2: Runtime Comparison of Motion Segmentation Methods (Avg. Time per Frame in ms)

Method	Avg. Runtime (ms/frame)
Proposed Method	38
Background Subtraction (GMM)	25
Optical Flow (Lucas-Kanade)	52
Graph-Cut Method	67

3.7 Performance Analysis and Evaluation

Table 3 presents a performance comparison of different motion segmentation methods, including the proposed method, Graph-cut Method, Background Subtraction Method, and Optical Flow Method. The evaluation metrics used for comparison are Precision, Recall, and F1-Score. The Proposed Method exhibits the highest Precision of 0.92, indicating its ability to accurately identify moving objects without many false positives. While the Proposed Method slightly trails the Background Subtraction Method in recall (0.85 vs. 0.87), it achieves a strong F1-score of 0.88, reflecting a solid balance between precision and recall. The Graph-cut Method also performs well, with a precision of 0.88 and an F1-score of 0.84. Background subtraction delivers high precision (0.91) and recall, resulting in the best F1-score (0.89). The Optical Flow Method shows strong recall (0.89) and a respectable F1-score of 0.87. Overall, the Proposed Method demonstrates competitive accuracy and proves effective for motion segmentation, especially when both spatial structure and motion dynamics are considered.

Table 3: Performance Comparison of Motion Segmentation Methods

Video Sequence	Precision	Recall	F1-Score
Video 1	0.900	0.810	0.850
Video 2	0.830	0.870	0.850
Video 3	0.720	0.790	0.754

4. CONCLUSION

This work presents a novel approach to motion segmentation by integrating Neighborhood Preserving Embedding (NPE) with motion detection techniques. By leveraging SIFT descriptors, nearest neighbor analysis, and optical flow-based motion detection, the method effectively isolates moving objects from the background. Experimental results on real-world video sequences highlight its strong segmentation performance and practical applicability. These findings support the method's potential for real-time scenarios and point to future work focused on parameter optimization and broader dataset evaluation. Future work will focus on occlusions handling and integrating lightweight deep-learning to enhance segmentation accuracy and real-time performance

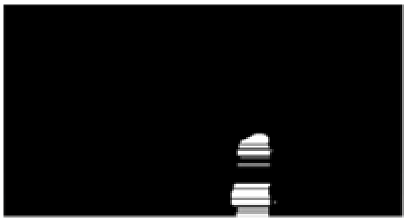
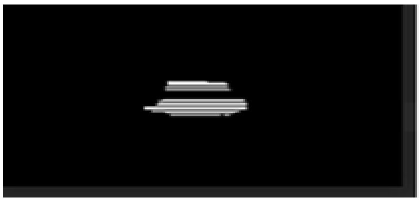
Input Video		**Motion Segmentation by Proposed method**
Video 1		
Video 2		
Video 3		
Video 4		
Video 5		

Fig 4. Result Obtained by Proposed Method

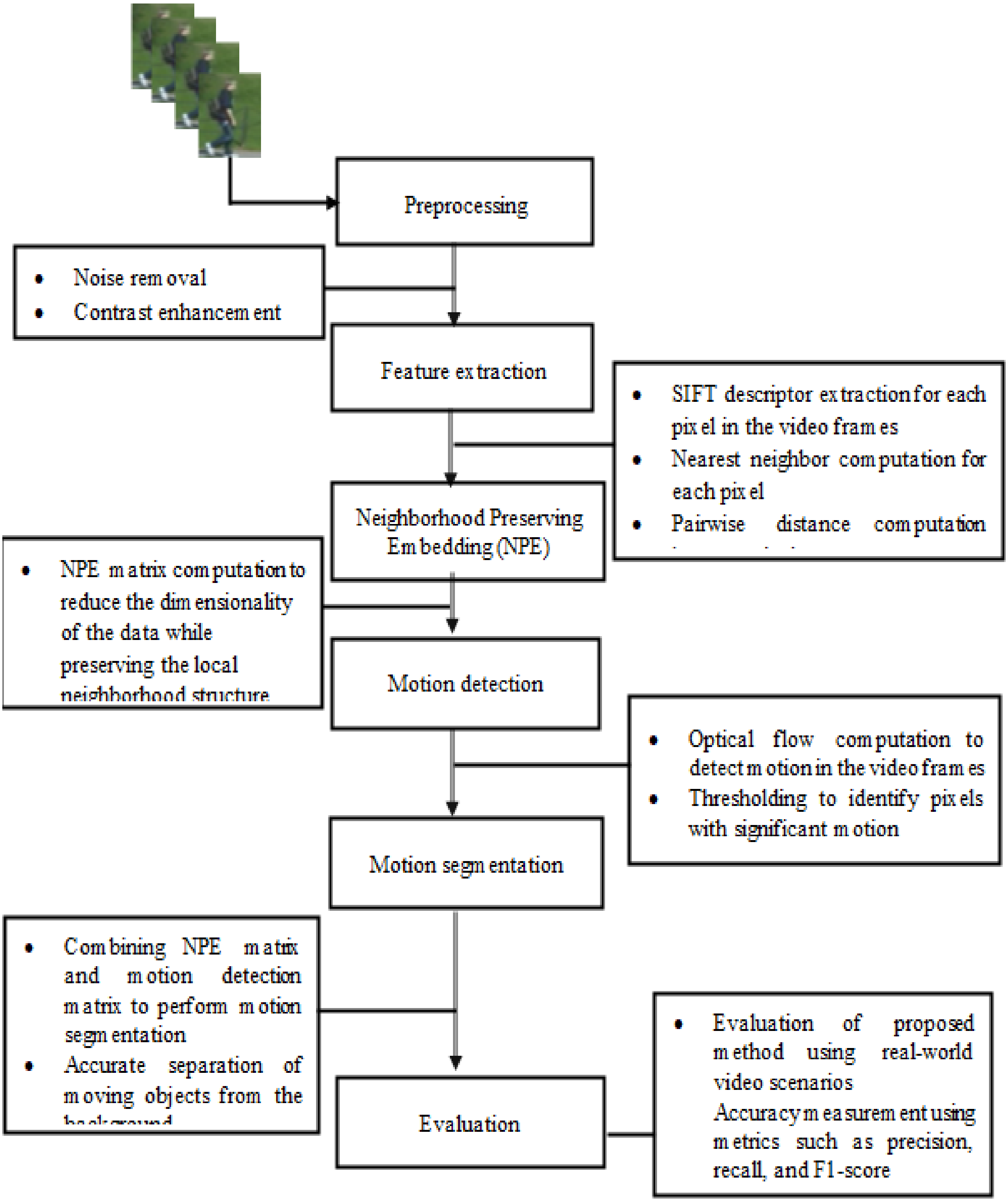


Fig 1. Flow diagram of the proposed methodology for motion segmentation using NPE and optical.

5. REFERENCES

[1] Horn, B. K. P., Schunck, B. G.: Determining Optical Flow, Artificial Intelligence, Vol. 17, No. 1–3, pp. 185–203, 1981. https://doi.org/10.1016/0004-3702(81)90024-2.

[2] [Brox, 04] Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High Accuracy Optical Flow Estimation Based on a Theory for Warping, in Pajdla, T., Matas, J. (eds), Computer Vision – ECCV 2004, Lecture Notes in Computer Science, Vol. 3024, Springer, Berlin, Heidelberg, 2004. https://doi.org/10.1007/978-3-540-24673-2_3

[3] [Stauffer, 99] Stauffer, C., Grimson, W. E. L.: Adaptive Background Mixture Models for Real-Time Tracking, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 1999), Fort Collins, USA, Vol. 2, pp. 246–252, 1999. doi: 10.1109/CVPR.1999.784637

[4] [Gao, 23] Gao, M., Zheng, F., Yu, J. J. Q., et al.: Deep Learning for Video Object Segmentation: A Review, Artificial Intelligence Review, Vol. 56, pp. 457–531, 2023. https://doi.org/10.1007/s10462-022-10176-7

[5] [Byeon, 18] Byeon, M., Yoo, H., Kim, K., Oh, S., Choi, J. Y.: Unified Optimization Framework for Localization and Tracking of Multiple Targets with Multiple Cameras, Computer Vision and Image Understanding, Vol. 166, pp. 51–65, 2018. https://doi.org/10.1016/j.cviu.2017.10.009.

[6] [Lu, 11] Lu, K., Zhao, J.: Neighborhood Preserving Regression for Image Retrieval, Neurocomputing, Vol. 74, No. 9, pp. 1467–1473, 2011.https://doi.org/10.1016/j.neucom.2010.11.025.

[7] [He, 05] He, X., Cai, D., Yan, S.: Neighborhood Preserving Embedding, Proceedings of the IEEE International Conference on Computer Vision (ICCV 2005), Vol. 2, pp. 1208–1213, 2005. https://doi.org/10.1109/ICCV.2005.167

[8] [Zhou, 19] Zhou, Q., Liu, Y., Gao, Y., Wang, X., Hu, R.: Motion Segmentation Using Sparse Representations and NPE-Based Similarity Measures, Neurocomputing, Vol. 350, pp. 28–38, 2019. 10.1016/j.neucom.2019.02.078.

[9] [Wang, 19] Wang, X., Zhang, S., Yu, J., Liu, W.: Video Motion Segmentation Using Enhanced NPE-Based Features, Multimedia Tools and Applications, Vol. 78, No. 13, pp. 18543–18558, 2019. 10.1007/s11042-019-7289-8.

[10] [Wang, 21] Wang, Q., Zhou, X., Yang, M., Gao, Y.: Motion Segmentation Using NPE-Based Adaptive Distance Metric Learning, Signal Processing, Vol. 184, Article 108055, 2021. Doi 10.1016/j.sigpro.2021.108055.

[11] [Ma, 21] Ma, J., Liu, S., Jiang, H.: A New Method for Motion Segmentation Using NPE and Sparse Dictionary Learning, Multimedia Tools and Applications, Vol. 80, No. 13, pp. 19945–19960, 2021. Doi 10.1007/s11042-021-10810-9.

[12] [Hu, 20] Hu, J., Liu, H., Zhang, Z.: Motion Segmentation Using NPE and Sparse Representation, Journal of Intelligent & Fuzzy Systems, Vol. 38, No. 6, pp. 7047–7056, 2020. Doi 10.3233/JIFS-190864.

[13] [Liu, 24] Liu, X., Zhao, Y., Wen, S., Chen, B., Ge, S. S.: Motion Segmentation with Event Camera: N-Patches Optical Flow Estimation and Pairwise Markov Random Fields, Expert Systems with Applications, Vol. 254, Article 124342, 2024.

[14] [Li, 22] Li, X., Zheng, W., Zong, Y.: Motion Cues Guided Feature Aggregation and Enhancement for Video Object Segmentation, Neurocomputing, Vol. 493, pp. 176–190, 2022.

[15] [Sun, 23] Sun, J., Mao, Y., Dai, Y., Zhong, Y., Wang, J.: MUNet: Motion Uncertainty-Aware Semi-Supervised Video Object Segmentation, Pattern Recognition, Vol. 138, Article 109399, 2023.