International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(7s) 2026 pp. 143-163
https://doi.org/10.70917/ijcisim-2026-3063
Article

Multi-Scale Spatio-Temporal Deformable Template Matching for Video Frame Extraction in Action Recognition

J. Usha Rani[1*], P. Raviraj[2]

[1,2] Department of Computer Science and Engineering, GSSS Institute of Engineering and Technology for Women, Mysuru, Affiliated to Visvesvaraya Technological University, Belagavi, India
Email: usharani@gsss.edu.in (J.U.R.); hodcse@gsss.edu.in (P.R.)
*usharani@gsss.edu.in

Abstract: —Video frame extraction has become an important component in action recognition and video analysis; however, its performance is often degraded by non-rigid object deformations, illumination variations, occlusions, background clutter, and temporal inconsistencies. To address these challenges, this research proposes a Multi-Scale Spatio-Temporal Deformable Template Matching (MSST-DTM) framework for robust and accurate video frame extraction. The proposed framework initially converts video sequences into normalized grayscale frames and applies oriented multi-scale Gaussian derivative filter banks to model discriminative structural information across different scales and orientations. Subsequently, a dual-stage deformable template matching strategy is implemented, integrating multi-level similarity estimation and edge-based control-point alignment to enhance spatial matching accuracy. Furthermore, flow-guided deformable convolution is incorporated to maintain temporal continuity between consecutive frames and improve key-frame selection reliability. The effectiveness of the proposed MSST-DTM framework is validated using the HMDB-51 and UCF-101 benchmark datasets. Experimental results demonstrate that the proposed MSST-DTM approach achieves an accuracy of 81.38% and 98.37% on HMDB-51 and UCF-101 datasets, respectively. Experimental results demonstrate that MSST-DTM provides a reliable and computationally efficient solution for video frame extraction and action recognition applications.
Keywords: —deformable template matching, dual-stage deformable template matching, multi-scale gaussian derivative filter, multi-scale spatio-temporal, object matching, traditional template matching

1. Introduction

Accurate frame extraction from video sequences is affected by motion variations, occlusions, illumination changes, and complex backgrounds [1]. To address these challenges, this object detection method utilizes spatial keypoints and region feature fusion to distinguish emotional categories from continuous sequences [2]. These techniques play a vital role in smart city development, where frame-sequence contrastive learning contributes to improved traffic management and enhanced public healthcare services [3]. However, extracting dynamic temporal features remains a primary challenge, as models often fail to leverage the rich prior knowledge available from mature static recognition datasets [4]. Industrial effectiveness monitoring has become more challenging due to blurry images and corrosive substances that adversely affect camera calibration performance [5]. Video event extraction (VEE) further extends event understanding to dynamic scenarios by capturing the natural temporal and causal relationships between actions and their associated arguments [6]. Deep Learning (DL) algorithms are frequently unreliable when evaluating unseen cases, requiring sophisticated techniques to assess prediction reliability in safety-critical applications [7]. Medical applications such as WCE analysis encounter challenges due to dark backgrounds and large borders within the image that hinder the detection [8]. Omnidirectional videos pose additional problems since uniform sampling does not account for important spatiotemporal degradations during the processing of high-resolution 360-degree videos [9]. The CBVR problem also suffers from predefined similarity measures and keyframe numbers that do not consider the unique properties of each video [10]. HBR involves capturing intricate spatiotemporal dependencies between visual features and temporal frames without assuming that all elements are of equal significance [11]. Object recognition in real-life scenarios poses particular problems because of the scarcity of large-scale databases capturing the variety and intricacy of natural conditions [12]. Finally, zero-shot video question answering is affected by background noise, thereby requiring a hierarchical filtering mechanism to identify the video segments relevant to the given question [13]. Object recognition plays an essential role in security; however, the computational expense of deep convolutional neural networks calls for optimization techniques to fine-tune hyperparameters in a constrained device environment [14]. The subtle motions of MEs require a double-branch network structure that can efficiently

capture local high-frequency spatial information and vertical motion information [15]. This inspires learning-based keyframe extraction based on semantic consistency for constructing global and local feature representation while minimizing the interference of noise frames [16]. To deal with class imbalance, high-resolution optical flow features coupled with multi-attention schemes have been explored to enhance discriminative regions [17]. Heterogeneous networks for both RGB and skeleton modality data facilitate multi-modal adaptation techniques for extracting highly complementary motion cues [18]. A multimodal fusion scheme incorporating appearance, geometric, and semantic information ensures reliable solutions for ME recognition in unconstrained environments [19]. Finally, the integration of global and local information ensures accurate recognition in unprocessed infrared surveillance videos while maintaining low computational overhead [20].

The Key highlights of this research are:

Template matching is a digital image processing method used to identify regions within an image that closely resemble a given template. Conventional template matching techniques work well for objects that are rigid, but they have trouble with changes in scale, shape, and deformation. This renders them unsuitable for object detection and recognition, because viewpoint modifications, and occlusions can cause non-rigid transformations. These issues are resolved by Multi-Scale Spatio-Temporal Deformable Template Matching (MSST-DTM), which enables the template to adjust to regional differences while maintaining its general structure. MSST-DTM uses elastic deformation models to allow for flexibility in matching, in contrast to rigid template matching, which depends on defined geometric transformations (such as translation, rotation, and scaling). This method increases robustness in real-world situations when objects change naturally. In video frame extraction and action recognition, MSST-DTM enhances performance by accurately identifying representative key frames from video sequences exhibiting complex motion patterns, viewpoint variations, background clutter, and non-rigid deformations. The proposed framework effectively captures both spatial structures and temporal dynamics through multi-scale deformable matching, enabling robust frame selection while preserving critical action-related information. By maintaining temporal consistency and adapting to motion variations across consecutive frames, MSST-DTM improves the quality of extracted key frames and facilitates more reliable action recognition. Consequently, the framework is most appropriate for different domains namely human action analysis, video surveillance, sports activity recognition, behavior understanding, and intelligent video retrieval systems.

The core concept of MSST-DTM is to match a deformable template (a model of the object you're trying to detect) to an image or dataset. Instead of rigidly matching the template, MSST-DTM allows the model to be deformed, enabling it to adapt to the shape and structure of the objects in the image. Recent advancements in Machine Learning (ML), DL, and optimization techniques have significantly improved performance and accuracy across various applications. An incorporation of Convolutional Neural Networks (CNNs), graph-based matching, and feature extraction techniques has enabled real-time, high-precision matching in large scale datasets. This paper explores the latest innovations in DTM, highlighting its applications, challenges, and prospects in object detection recognition.

The significant innovation of this research is presented in the unified MSST-DTM pipeline, which incorporates the oriented Gaussian derivative filtering, dual-stage deformable matching, and flow-guided deformable convolution. Unlike existing existing approaches, which use these advancements independently, the proposed approach integrates them within an individual optimized framework for enhancing both spatial matching accuracy and temporal frame reliability.

The rest of the paper is arranged as follows: Section 2 presents the literature survey. Section 3 specifies the DTM. Section 4 illustrates the experimental results. Section 5 discusses the conclusion.

2. LITERATURE SURVEY

Now a days, recent developments have been introduced in action recognition, especially obtained through developments in DL approaches. This portion concentrates on determining the selection of most important approaches in action recognition. The video contains higher amount of complexity because of an occurrence of temporal and spatial features as well as their consistent time dependances. To efficiently identify the video data, urbane Deep Neural Networks (DNN) was introduced for feature extraction and understanding the temporal dependencies. M. Jayamohan and S. Yuvaraj [21] presented the SegNet approach through integrated attention unit and improved Bidirectional Gated Recurrent Unit (BiGRU) approach for human action recognition through semantic image segmentation. In the presented framework, the approach employed the graded feature extraction from input image through encoder, while the decoder concentrated on redeveloping the segmented result. The attention was applied to encoded feature maps, augmented an approach's capacity for modeling the representations across large spatial ranges. The BiGRU layer was performed to model sequential representations in fused feature maps. An incorporation of SegNet approach through attention mechanism and BiGRU, featuring the encoder-decoder framework for feature extraction, segmentation and classification, illustrating its effectiveness in

modeling nuanced spatiotemporal features. Davar Giveki [22] developed the new Gated Recurrent unit (GRU) approach to understand the important features through spatial data and time representations in action recognition. Furthermore, the new DNN approach was introduced by new GRU method for human action recognition. To illustrate the functionality and applicability of developed approach in addressing real-time issues, an engine block gathering dataset was acquired and effectiveness of introduced approach was estimated on this dataset.

Guancheng Huang et al. [23] introduced the spatiotemporal feature enhancement approach for action recognition. Primarily, the new temporal feature incorporation model was designed for improving short-range temporal features and improve action-based features. After that, the group convolution superposition component was presented to improve the approachable temporal features area and enhance the understanding capability of an approach to long-term spatial and temporal features. Hao Pan et al. [24] presented the action recognition approach related to lightweight framework and rough-to-fine keyframe extraction. The presented approach comprised of different modules: An initial module developed the keyframe extraction pipeline related to grayscale and feature descriptors and performed the rough-fine concept for the video keyframe extraction. The second module presented the attention-based feature extraction pipeline, that incorporated the decoupling ideas through attention mechanisms for the model performance improvement and minimized the network parameters. The final module was an enhanced attention module that optimized the local data representation. Eventually, the addition of residual modules facilitates the fusion of feature information across different convolutional layers. M. Jayamohan and S. Yuvaraj [25] presented the new action recognition approach which incorporated the attention mechanism through modified GRU. The presented approach utilized the InceptionV3 for feature extraction, that was integrated through the attention mechanisms which concentrated on most complex areas of input sequences. This approach enables better determination of the most important video perspectives, thus improved the precision of action recognition. The attention-improved features were further performed through modified GRU, that utilized an attention mechanism to classify video behaviors most efficiently, especially for large video sequences.

Shaimaa Yosry et al. [26] suggested the new multi-stream framework designed with different approaches fused through various fusion methods. Initial method integrated the CNNs in two-dimensional (2D-CNNs) with Long Short-Term Memory (LSTM) method to understand long-term temporal and spatial features for recognition of human action. Another method was three-dimensional CNNs (3D-CNNs) which collected rapid spatial-temporal features from video frames. Followingly, different architectures are state to describe how different fusion structure enhance the action recognition effectiveness. This study estimated the approaches through early and late fusion, whereas late-fusion identified the cause of timely feature fusion of different approaches for action recognition. This different integration approaches identified how much every temporal and feature feature impact of an approach's performance. Rukiye Savran Kızıltepe et al. [27] implemented the comparative analysis for investigating how temporal data can be utilized for enhancing the effectiveness of video classification when the CNNs and Recurrent Neural Networks (RNNs) were incorporated in different pipelines. To improve the effectiveness of the proposed framework through efficient integration of CNNs and RNN models, a novel action template-assisted keyframe extraction approach was developed by identifying the informative regions in each frame and selecting keyframes based on the similarity between those areas. Elaheh Dastbaravardeh et al. [28] introduced the enhanced approach for human action detection in minimum-size and low-resolution videos through performing the CNNs through Autoencoders (AEs) and Channel Attention Mechanisms (CAMs). Through improving the blocks through most demonstrative features, convolutional layers extracted the discriminative features from different models. Moreover, feature's arbitrary sampling was utilized prior significant processing to enhance the accuracy whereas performing minimum data. The objective is to enhance performance while addressing these challenges namely computational complexity, uncertainty and overfitting through leveraging CNN-CAM and AE.

Recent self-supervised learning approaches have illustrated effective performance in video frame extraction and temporal representation understanding through learning without manual labels. Nevertheless, existing methods typically need large-scale pretraining datasets and high computational cost. Moreover, recent transformer-based approaches, such as the Video Swin Transformer, attain existing temporal consistency in video analysis. However, these approaches need large-scale labeled datasets and high computational resources. On the other hand, the proposed MSST-DTM pipeline is introduced for computationally constrained environments and operates efficiently through only grayscale video sequences and constrained training data. The proposed MSST-DTM instead focuses on a lightweight and interpretable framework suitable for small datasets and real-time applications.

3. PROPOSED METHODOLOGY

The proposed MSST-DTM approach is designed to perform robust video frame extraction by jointly exploiting spatial feature representation and temporal motion consistency. The complete architecture consists of four major stages: video preprocessing, multi-scale feature extraction, DTM, and temporal refinement. Primarily,

the input video sequence is decomposed into individual frames and transformed into normalized grayscale images to minimize computational complexity and illumination sensitivity. Subsequently, oriented multi-scale Gaussian derivative filter banks are performed for extracting the discriminative edge and structural features from each frame. Subsequently, these features are processed using a dual-stage deformable matching mechanism which integrates coarse-to-fine similarity estimation with edge-based control-point alignment to precisely determine corresponding regions despite geometric deformations. To further improve temporal coherence between neighbouring frames, flow-guided deformable convolution is integrated to adaptively enhance the motion variations and refine frame correspondences. The final output consists of informative key frames which maintain both spatial structure and temporal dynamics, thus improving the effectiveness of subsequent video analysis and action recognition tasks. Fig. 1 specifies the algorithmic Framework for MSST-DTM in action recognition.

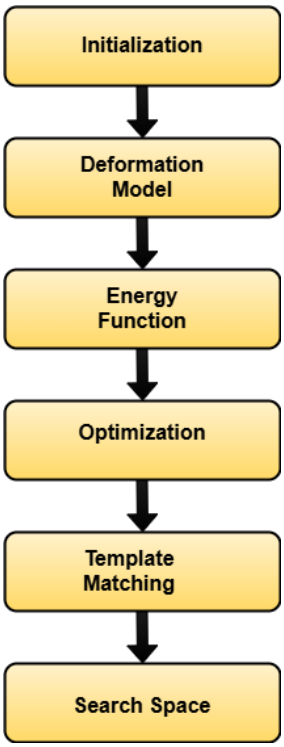

Fig. 1. Algorithmic framework for MSST-DTM in video frame extraction in action recognition.

A. Dataset

This research considers the different emotion recognition datasets such as HMDB-51 [29] and UCF-101 [30] to validate the effectiveness of proposed method. The UCF-101 offers a general dataset of 13,320 video clips, which are categorized as 101 human activity classes. These classes span a broad range of actions such as physical activities, sports, communications with substances, musical effectiveness, and interactive actions. The videos, obtained mainly from YouTube, represent crucial differences in components namely environmental backgrounds, object appearances, and camera movements, presenting the complex data for performing on action recognition. Furthermore, the dataset arranges the videos into 25 groups, where every group involves four to seven clips, which transmit the same viewpoint or backgrounds, offering the extra layer of structure for experimental designs.

The HMDB-51 dataset features 6766 video clips partitioned as 51 action classes, every class involving a atleast 101 samples. This benchmark obtains the videos YouTube and archival acquisition, maintaining a different representative of acts and actions. Furthermore, it provides comprehensive annotations, requiring different features namely camera perception, incidence or nonappearance of motion, number of actors and video quality involved. Table I illustrates the characteristics of UCF-101 and HMDB-51 datasets for human action recognition.

TABLE I. CHARACTERISTICS OF UCF-101 AND HMDB-51 DATASETS FOR HUMAN ACTION RECOGNITION

Characteristics	HMDB-51	UCF-101
Number of classes	51	101
Video quantity	6766	13320
Videos duration	3-6s	4-10s

Challenges faced	Pose discrepancies, occlusions, object variability	Varied camera conditions, illumination shifts

B. Preprocessing

In different environments modeled on action recognition, this research utilized the DL approach for recognizing the human actions and detect the abnormal behaviour or unusual activities. In terms of improving the model performance, DL approaches are most capable to handle the large datasets. The procedure of preprocessing comprises of eliminating the frames from captured videos in earlier. The subfolder termed after every video is recognized as well as handles along with the frames. The JPG images are designed from video frames. The data are resized into 224×224 dimensions. Previously being kept in the folder, the testing video is also converted into frames and scaled to 224×224. The bilinear approach is utilized for low, medium and high-resolution images. For downsampling the images, the fast minimization of dimensions is preferred. Then, the random sampling is utilized for generating some frames in an action video. Through utilizing the frame sampling for minimizing video size, unimportant data processing is saved. According to data features, various videos have distinct number of shot segments. With respect to minimize the image counts available for every segment, arbitrary select one frame here.

After the video frames are preprocessed into grayscale, the frame selection strategy is applied to reduce the idleness and keeps only the most informative [31]. This method crucially minimizes the computational costs and maintains that the network processes only the frames demonstrating key temporal transitions, thus improving the analysis. This strategy crucially minimizes the computational costs and maintains that the effectiveness of the action recognition pipeline.

C. Deformable Template Matching (DTM)

DTM is a powerful technique used for pattern recognition, image matching, and object detection, especially when dealing with non-rigid or deformable objects. It's often applied in computer vision, medical imaging, and shape matching problems where the target shapes or patterns may not be rigid and can undergo transformations such as translation, rotation, scaling, and even non-rigid deformations like bending or stretching. DTM is an advanced computer vision technique used for object detection and extraction. As compared to traditional template matching methods that rely on rigid templates, DTM accommodates shape deformations, thereby enhancing robustness against variations in object appearance caused by pose changes, occlusions, and distortions distortions. The core concept of MSST-DTM is to match a deformable template to an image or dataset. Instead of rigidly matching the template, MSST-DTM allows the model to be deformed, enabling it to adapt to the shape and structure of the objects in the image. The flowchart illustrates the core steps involved in the DTM algorithm used for image analysis and pattern recognition. The process begins with initialization, where a predefined template is created to represent the object of interest. This template can be a shape, curve, or a set of key points and may be derived either manually or through automatic processes. Next, a Deformation Model is applied, allowing the template to adapt to variations in the target object. The model accounts for both rigid transformations (like scaling, translation, and rotation) and non-rigid elastic deformations such as bending and stretching. The energy function is then defined to guide the matching process. This function combines two components: a matching energy that evaluates how well the deformed template fits the target image and a regularization energy that ensures smooth and realistic deformations. An aim is to reduce the total energy for optimal alignment. Optimization techniques, such as gradient descent or dynamic programming, are used to adjust the template iteratively until the best match is found. After optimization, the template matching phase compares the deformed template with the corresponding region in the target image. A close match with minimal deformation indicates a likely detection of the target object. The algorithm operates within a Search Space that includes both spatial and deformation parameters. Efficient search methods ensure that the best possible match is found without exhaustive computation. This structured approach allows MSST-DTM to accurately detect and align objects in noisy or complex images by combining flexibility with mathematical rigor. Fig. 2 specifies the architecture of MSST-DTM for video frame extraction, including a scale-adaptive deep convolutional feature extraction approach and FMCC-based comparison measurement framework.

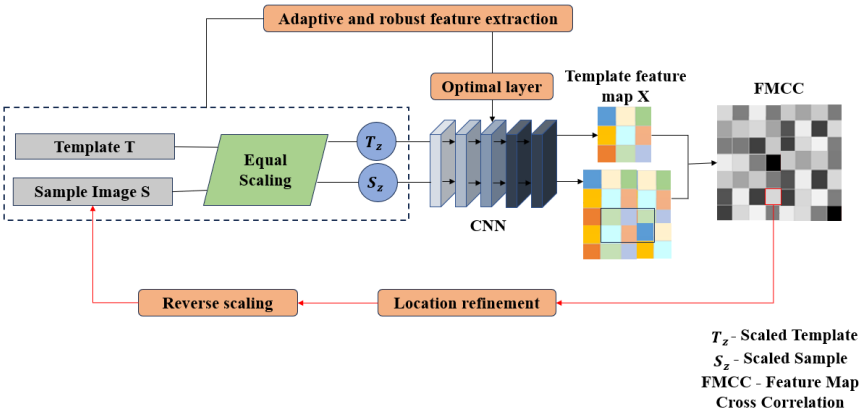

Fig. 2. Architecture of MSST-DTM for video frame extraction, involving a scale-adaptive deep convolutional feature extraction approach and FMCC-based resemblance estimation framework.

1) Feature Extraction of video frames

Provided the different template image T and source identification image I, the proposed method purposes to evaluate the geometric transformation among T and I for offering an accurate position and object's pose in an image I. For scenes containing different activities and applicant templates, a coarse phase first estimates an initial homography to select the appropriate template, followed by a refinement stage for precise localization. Unlike conventional template matching, this task involves a domain gap, as T is a different mask while I demonstrates grayscale image. Moreover, grayscale images may contain strong reflections, making photometric similarity-based matching ineffective. To overcome these challenges, a translation network converts both images into edge maps using a lightweight edge detector. This transformation ensures domain consistency and enables reliable feature.

This research uses the most common architecture as like to SuperPoint for feature extraction at various scales from both images after conversion. SuperPoint contains VGG-style encoder trained through self-supervision and demonstrates improved model performance in different vision tasks. This research keeps an encoder model of SuperPoint as the local feature extraction network. Provided an input image of size $H \times W$, the feature extraction module generates feature maps at various resolutions; keep second layer feature map ($\hat{F} \in RH/2 \times W/2 \times D$) and final layer feature map ($\tilde{F} \in RH/8 \times W/8 \times C$). Hence, $\tilde{F}T$ and $\tilde{F}I$ are uneven level features, $\hat{F}T$ and $\hat{F}\ I$ are fine-level features.

2) Establishing Coarse Matches

This research establishes the coarse matches through transformed features $\tilde{F}_{tr}^{T}$ and $\tilde{F}_{tr}^{I}$. The improved transport layer is acquired as the separable matching layer and the score matrix S is primarily estimated through dot-product similarity of transformed features, which is formulated in Eq. (1) as follows:

$$S(i,j) = \langle \tilde{F}_{tr}^{T}(i), \tilde{F}_{tr}^{I} \rangle \qquad (1)$$

This score matrix S is utilized a cost matrix in an incomplete task issue and it is tackled through Sinkhorn approach to acquire the confidence assignment matrix C. To acquire the most consistent matches, Mutual Nearest Neighbour (MNN) standard is imposed, and identical pairs through confidence values are greater than threshold θ_c are maintained. The group of course-level matches M_c is formulated in Eq. (2) as follows:

$$M_c = \{(i,j) | \forall (i,j) \in MNN(C), C(i,j) \geq \theta_c\} \qquad (2)$$

Alternative identical layer methods leverage a dual-softmax operator, which adopts the softmax to both dimensions of S to acquire the likelihood of MNN.

3) Confidence weights related to spatial consistency

A derivable matching layer offers the uncertain match group M_c related to feature dot-product resemblance. In this manner, various mismatched points are incorrectly observed as matching pairs because of appearance similarity. To ignore this, this research incorporated the new constraint related to the remark which template matching has important property such as accurate correspondences have same geometric transformations, whereas the transformations of outliers are arbitrary. On the other hand, the proposed approach assumes the non-rigid deformation in which the pairwise distance among far points are most similar to different than among closer ones. Thus, the spectral matching is improved and developed the approach related to distance-and-angle reliability for non-rigid deformation outlier rejection.

Assume β specifies the distance consistency term estimating the modification in matched pair's length. To accommodate scale variations, the distance among the matching points are normalized on template and image individually. After that, for different coarse matches $a = (i, i')$ and $b = (i, i')$, β is described in Eq. (3) as follows:

$$\beta_{(a,b)} = \left[1 - \left(\frac{d_{ij}}{d_{i'j'}} - 1\right)^2 / \sigma_d^2\right]_+ \quad (3)$$

Whereas, d_{ij} denotes normalized pairwise distance among i and j, $[.]_+$ and σ_d illustrates distance parameter controlling compassion to variations in appropriate length. The modifications in directions are penalized through the triplet-wise angle. This research estimates the pointed suitability from coarse feature points. For identical pair $a = (i, j)$ with positions p_i and p_j, primarily select the K-nearest Neighbour (KNN) N_i of p_i. To enhance the robustness, the maximum value c_{ij} among KNN are chosen as the angle assets for matching pair (i, j). As for distance consistency β, the angular compatibility α is formulated in Eq. (4) as follows:

$$\alpha_{(a,b)} = \left[1 - \left(\frac{c_{ij}}{c_{i'j'}} - 1\right)^2 / \sigma_\alpha^2\right]_+ \quad (4)$$

The eventual spatial consistency of matches a and b is formulated in Eq. (5) as follows:

$$E(a, b) = \lambda_c \alpha_{(a,b)} + (1 - \lambda_c)\beta_{(a,b)} \quad (5)$$

Where, λ_c illustrates the control weight, $E(a, b)$ denotes large only of the different communications a and b are extremely spatially compatible. An important eigenvector of e of consistency matrix E is observed as the valid point's likelihood of matches.

D. Advantages of DTM

- **Robustness**: MSST-DTM allows for flexible matching, making it more robust to variations in scale, rotation, and shape deformation.
- **Real-World Applicability**: Since most objects in real world undergo non-rigid transformations, MSST-DTM is better suited to real-world applications than rigid template matching.
- **Detailed Matching**: It can capture subtle variations in shapes and patterns, which makes it more precise for complex shapes.

E. Challenges

- **Computational Complexity**: Due to the large search space and the need for optimization, DTM can be computationally expensive.
- **Overfitting**: There is a risk of the template becoming too specific to a particular instance of the object.
- **Deformation Model**: Defining the deformation model properly can be challenging, as it needs to strike a balance between flexibility and accuracy.

The proposed MSST-DTM framework is organised into four following phases. Initially, an input video is decomposed into individual frames and converted into normalized grayscale representations. Next, oriented multi-scale Gaussian derivative filter banks are used for extracting the robust local structural features across multiple scales and orientations. Then, the DTM is employed through different complementary apparatuses: multi-level similarity estimation and edge-based control-point alignment. Eventually, flow-guided deformable convolution is performed for refining the temporal continuity between consecutive frames and to produce final extracted key frames.

Algorithm 1: Pseudocode of the proposed MSST-DTM approach for video frame extraction in action recognition

Input:

template ← Predefined template image

target_image ← Image to be matched

deformation_model ← Allowed deformation parameters

```
Output:
    match_position ← Location of best match in target_image

begin
    // Initialization
    deformation_set ← generate_possible_deformations(deformation_model)
    min_energy ← ∞
    best_deformation ← null

    // Deformation Optimization
    foreach deformation ∈ deformation_set do
        matching_energy ← compute_matching_energy(template, target_image, deformation)
        smoothness_energy ← compute_smoothness_energy(deformation)
        total_energy ← matching_energy + smoothness_energy

        if total_energy < min_energy then
            min_energy ← total_energy
            best_deformation ← deformation
        end if
    end foreach

    // Apply Optimal Deformation
    deformed_template ← apply_deformation(template, best_deformation)

    // Template Matching
    match_position ← find_best_match(deformed_template, target_image)

    // Output Result
    if match_position ≠ null then
        print("Template matched at position: ", match_position)
    else
        print("No match found")
    end if
end
```

Explanation of the MSST-DTM:

1. Initialize the Template Model: Load the predefined template that represents the object to be matched. Define the deformation model, which specifies the rules and constraints for allowable deformations.

2. Define the Energy Function: Create an energy function that combines two components: o Matching Energy: Measures how well the deformed template matches the target image. o Smoothness Energy: Ensures that the deformation is smooth and realistic.
3. Optimization Process: Iterate through possible deformations defined by the deformation model. For each deformation, compute the total energy and select the one with the lowest energy, indicating the best match.
4. Apply the Best Deformation: Transform the template using the best deformation found in the previous step.
5. Match the Deformed Template: Search for the position in the target image where the deformed template best matches.
6. Output the Result: If a match is found, output the position; otherwise, indicate that no match was found. This pseudocode provides a structured approach to implementing DTM, focusing on the initialization, energy computation, optimization, and matching processes.

F. Steps to Extract Images from Video Using DTM

1. Load the Video

- Open the video file using a video processing library like OpenCV (for Python).
- Extract frames from the video.

2. Define the Template

- Define a template (model) of the object you want to track or match. The template could be an image or a set of key points representing the object.
- This template could either be predefined or dynamically selected.

3. Preprocess the Video Frames

- Convert the video frames to grayscale or another format for easier processing (optional but often used in template matching).
- Resize or adjust frames if necessary to match the template dimensions.

4. Apply DTM to Each Frame

- For each frame, apply the DTM algorithm to find the best match.This step involves:
- Initializing the deformation model (e.g., elastic deformation, affine transformation).
- Using an energy function that combines matching energy and smoothness energy (as outlined in the previous pseudocode).
- Optimizing the deformation to find the best match (by minimizing the energy function).
- Deform the template to adapt to the object in the current frame.

5. Extract Frames with a Good Match

- Once you get the deformation that minimizes the energy, you can consider the match valid if the deformation is within an acceptable range (i.e., not too large, meaning the object is well-aligned with the template).
- Save or process the frames where the template match is good.

6. Post-processing (optional)

- You might want to apply a threshold to the matching score or deformation parameters to determine which frames are considered good matches.
- Optionally, save or display those frames for further analysis or output them as images.

7. Video Representation:

Represent the input video as a sequence of image frames in Eq. (6) as follows:

$$V = \{I_t\}\frac{T}{t=1} \qquad (6)$$

Where I_t is the image at time/frame t, and each image has dimensions H×W(height by width).

8. Deformable Template Definition:

Define a deformable template $T_m(x;\theta)$,

Where: θ are the base parameters of the template (e.g., shape or intensity profile), x are deformation parameters (e.g., translation, scaling, bending, or local displacements).

9. Matching Criterion:

Use a similarity function $L\big(I_t, T_m(x;\theta)\big)$ that compares the deformed template to the frame I_t. This function typically computes a loss or energy (such as squared difference, mutual information, or edge alignment score). The dynamic template adaptation is trained through the following objective function, which is formulated in Eq. (7) as follows:

$$L_{Total} = \lambda_1 L_{rec} + \lambda_2 L_{smooth} + \lambda_3 L_{reg} \qquad (7)$$

Where, L_{rec} specifies the frame reconstruction loss; L_{smooth} enforces temporal consistency among neighbouring flow vectors; L_{reg} denotes penalizes excessive deformation offsets. In the experiments, $\lambda_1 = 1.0$, $\lambda_1 = 0.1$ and $\lambda_1 = 0.01$ individually. To improve robustness against illumination variation, the similarity score is computed after local contrast normalization, which is formulated in Eq. (8) as follows:

$$S(I,T) = \frac{(I-\mu_I)(T-\mu_I)}{\sigma_I \sigma_T} \qquad (8)$$

Where, μ and σ denotes the local mean and standard deviation. This normalization reduces the influence of sudden brightness fluctuations and improves matching stability under flash or low-light conditions.

10. Optimal Deformation Search:

For each frame t, the optimal deformation parameters x_t^* are determined by minimizing the mismatch between the frame and the template in Eq. (9) as follows:

$$x_t^* = argL\big(I_t, T_m(x;\theta)\big) \qquad (9)$$

11. Region Extraction:

Using the optimal deformation x_t^*, extract the best-matching region from each frame. Let ϕ $(I_t x_t^*)$ denote the operation that extracts or aligns the template to the image region. Then, the extracted image sequence is shown in Eq. (10) as follows:

$$\varepsilon = \{R_t = \phi(I_t x_t^*)\}\frac{T}{t=1} \qquad (10)$$

This process is typically used in medical imaging, object tracking, and biometric recognition, where objects can deform or move across video frames.

G. Deformation Models

The deformation model can include affine transformations (rotation, scaling, translation) or more complex non-rigid deformations (elastic deformations). The selection of deformation approach relies on the object being tracked and flexibility needed in matching.

- Optimization: The optimization step is crucial for finding the best match. Depending on the complexity of the deformations and the video, you may need to use sophisticated optimization algorithms like gradient descent or dynamic programming.
- Thresholding for Good Match: Define a threshold on the energy function or deformation to determine when a match is good enough. This ensures that you don't mistakenly extract frames with poor matches.

Algorithm 2: Extracting images from video

Input:

- Video file V
- Deformable template T_m
- Deformation model $\mathcal{M}$

Output:

- Set of extracted frames Ɛ

1. Load the video file:

Initialize video stream V ← OpenVideo(video_path)

2. Load or define the deformable template:

T_m ← LoadTemplate()

3. Define the deformation model:

$\mathcal{M}$ ← DefineDeformationModel()

4. Initialize variables:

Ɛ ⟵ ∅ (set of extracted frames)
t ← 0 (frame index)

5. While video has more frames, do:

a. Read next frame:
   I_t ← ReadFrame(V)
   If I_t = ∅, then break
b. Preprocess the frame:
   I_t^gray ← Preprocess(I_t)
c. Perform DTM:
   x_t* ← argmin_x L(I_t^gray, T_m(x; $\mathcal{M}$))
d. Evaluate match quality:
   If IsGoodMatch(x_t*), then:
      ε ← Ɛ ∪ {I_t}
e. Increment frame index:
   t ← t + 1

6. Release video resources:

CloseVideo(V)

H. Proposed enhanced matching algorithm

The DTM process begins with two input images: a source image I_1 and a target image I_2 The goal is to align the template derived from I_1 onto I_2 by computing an optimal deformation field t that minimizes a matching cost c.

The algorithm proceeds as follows:

1. Decomposition:

The source image I_1 is decomposed into a grid of sub-regions or patches based on specified dimensions n×m times. This step results in a set of local template patches and their positions, denoted as p

2. Initialization:

An initial deformation field t is established over the target image I_2. The field contains transformation parameters (such as position, scale, or orientation) for each corresponding sub-region defined by the grid.

3. Scale Setting:

A scale factor s is initialized to control the level of allowable deformation during each iteration. This factor will incrementally increase to explore wider deformation possibilities.

4. Iterative Optimization:

The algorithm enters a loop that iteratively refines the deformation field t. At the beginning of each iteration, the current deformation field is stored as a reference (t_{old}) and a cost accumulator c is reset.

For each grid location (i,j) the algorithm evaluates the optimal local deformation by invoking a cost-minimization function (denoted as MinCdtm), which computes both the updated transformation at that point and the associated cost. This cost is added to the total cost c for the current iteration.

5. Convergence Check:

After processing all sub-regions, the updated deformation field t is compared to its previous state t_{old}.If no changes are observed, the algorithm concludes that convergence has been reached and terminates.

6. Scale Increment:

If convergence is not achieved, the scale factor s is incremented to allow for broader deformation flexibility, and the process repeats.

This proposed algorithm of DTM is an iterative optimization.

4. EXPERIMENTAL RESULTS

All experiments employed through python with Intel Core i7-12700 processor, 32 GB RAM, and an NVIDIA RTX 3060 GPU with 12 GB memory. Under this configuration, the proposed MSST-DTM achieved an average speed of 14 FPS for 640 × 480 video sequences. In this research, the proposed MSST-DTM method is tested and demonstrated using different standard datasets. Moreover, this research compares the proposed method with existing approaches and provides performance analysis across all datasets used. Each dataset is partitioned into two portions, such as 80% of the data is used for training and 20% for testing, split at the video level, ensuring that video identities are mutually exclusive across the two sets. In this research, the comparison results are categorized into two groups for clarity. Tables II–VII present reproduced baseline models evaluated under the same preprocessing, training, and testing conditions as the proposed method. In contrast, Table VIII summarizes results reported in that the models were reimplemented and evaluated under a unified experimental protocol to ensure a fair and unbiased comparison with the proposed MSST-DTM framework. This distinction ensures fair comparison and improves transparency in performance evaluation. In section 4.1, baseline methods values are reproduced based on the proposed method settings. In section 4.2, the values for the existing methods are taken from the respective literature papers. The proposed method's significance is validated utilizing the range of performance metrics, namely F1-score, recall, precision and accuracy. The mathematical expressions are formulated in Eqs. (11) to (14).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100 \quad (11)$$

$$Precision = \frac{TP}{TP+FP} \times 100 \quad (12)$$

$$Recall = \frac{TP}{TP+FN} \times 100 \quad (13)$$

$$F1-score = \frac{2TP}{2TP+FP+FN} \times 100 \quad (14)$$

Where FP, TP, FN and TN represents False Positive, True Positive, False Negative and True Negative.The dataset is divided into 80:20 training and testing sets. The dropout rate is set to 0.3, the batch size is 32, and training is performed for 100 epochs using the Adam optimizer with a learning rate of 0.001. Early stopping with a patience value 10 is used to prevent overfitting. Adam is applied for hyperparameter optimization over learning rate, batch size, hidden units, and dropout configurations.

A. Performance estimation

The performance of the proposed approach with other approches is demonstrated using various datasets. The proposed MSST-DTM method is validated using existing methods, namely GRU, RNN, LSTM, and BiLSTM. The proposed MSST-DTM framework improves frame extraction performance by exploiting deformable spatial correspondence, multi-scale structural information, and temporal consistency. These characteristics enable the framework to identify representative key frames more effectively than sequence-based baseline methods.

1) Video Frame Intensity Signal over Time

Fig. 2 presents a time-series plot of pixel intensity values extracted from a video signal, representing a 10-second slice of data. The x-axis denotes Time (ms), spanning from 0 to 10,000 milliseconds, while the y-axis demonstrates the respective intensity values in arbitrary units.

The signal exhibits a periodic pattern characterized by:

- Sharp peaks at regular intervals, suggesting the presence of highly dynamic or transient visual events in the video (e.g., motion, flashes, or object transitions),
- Low-frequency modulations between peaks, reflecting gradual changes in visual content or background information,
- A baseline variation between approximately 480 and 530 unit indicating stable yet slightly fluctuating luminance levels in the video content.

These features imply that the video contains cyclic or rhythmic events, potentially useful for tasks such as motion segmentation, heartbeat or respiration monitoring, or periodic template matching. The spike regularity may also support frame synchronization, temporal localization, or trigger-based frame extraction for further analysis using techniques like DTM.

Fig. 3 presents a 10-second temporal slice of a signal extracted from sequential video frames. The x-axis illustrates time in milliseconds, while the y-axis denotes random unit, reflecting the magnitude of a computed feature—such as edge intensity, motion energy, or pixel variation—from the video. The signal exhibits periodic peaks approximately every 800–1000 milliseconds, suggesting the presence of regular visual or motion-related events in the recorded footage. These prominent peaks may correspond to heartbeat-like pulses or cyclic motion patterns, making them suitable anchors for synchronization or segmentation in video analysis. The consistent repetition in signal structure serves as a foundational basis for subsequent DTM and pattern recognition algorithms, enabling robust identification and alignment of key frames. This time-domain representation underscores the temporal stability and periodic nature of the visual activity within the selected video segment.

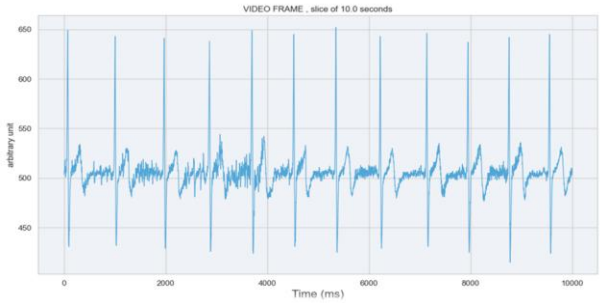

Fig. 3. Temporal signal analysis from video frame sequences.

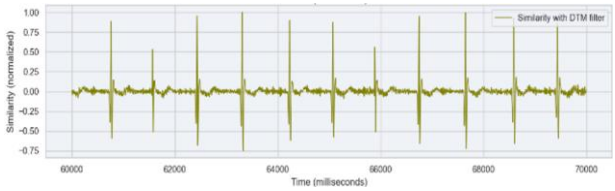

Fig. 4. Event detection using DTM.

The results depicted in Fig. 4 demonstrate the effectiveness of the DTM algorithm in accurately detecting recurring temporal events within a dynamic video signal. The upper plot shows the video signal amplitude over time, with distinct peaks representing the occurrence of periodic events. These peaks were successfully detected through image slicing, as indicated by the orange markers.

In the lower plot, the similarity response generated by the MSST-DTM filter is presented. High similarity peaks align consistently with the detected video peaks, suggesting that the MSST-DTM method reliably captures the underlying periodic structure in the data. The sharpness and periodicity of these similarity spikes reflect the robustness of the algorithm against noise and variability in signal shape, further confirming its suitability for time-localized pattern detection.

Overall, the figure validates that MSST-DTM not only enhances event detection accuracy but also offers strong temporal resolution, making it well-suited for analyzing video signals where object appearance changes subtly over time.

Fig. 5 specifies the grouping of shape peaks using MSST-DTM filtering. The blue curve represents the template-matched signal. Orange dots indicate samples exceeding the similarity threshold, corresponding to detected high-confidence matches. The black triangle marks the median position of the grouped peaks, providing a stable and accurate estimate of the event location.

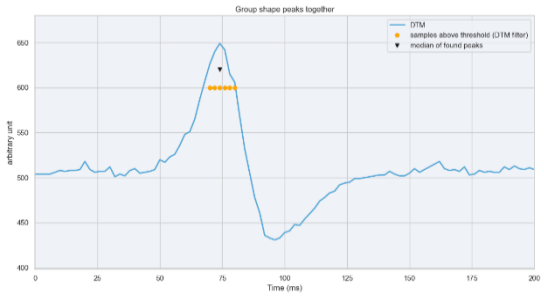

Fig. 5. Grouping of shape peaks using MSST-DTM filtering.

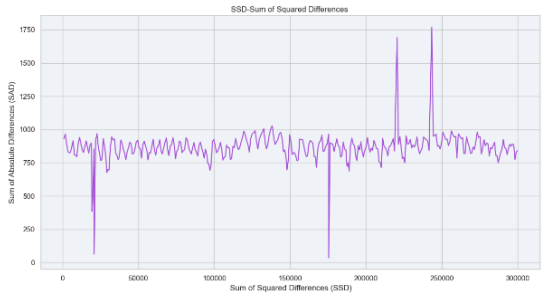

Fig. 6. The relationship between the sum of squared differences (SSD) and the sum of absolute differences (SAD).

Fig. 6 illustrates the relationship between Sum of Squared Differences (SSD) and Sum of Absolute Differences (SAD). These metrics are commonly used in image processing, computer vision, and video compression — particularly for tasks like template matching, motion estimation, or block matching.

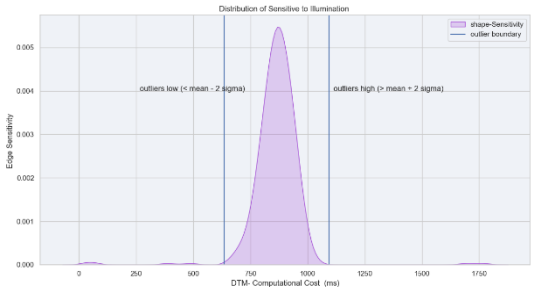

Fig. 7. Distribution plot showing how a variable referred to as "shape Sensitivity" (likely to illumination) is distributed over MSST-DTM computational Cost (ms).

Fig. 7 illustrates the distribution plot showing how a variable referred to as "shape Sensitivity" (likely to illumination) is distributed over MSST-DTM Computational Cost (ms). It includes statistical boundaries to highlight outliers.

Fig. 8 illustrates a comparative analysis between DTM and Rigid Matching after correction plotted as a function of direct pixel position along x-axis. The y-axis specifies a matching metric, labeled as Rigid Matching, which likely corresponds to an alignment or similarity score derived from pixel intensities or spatial correspondence.

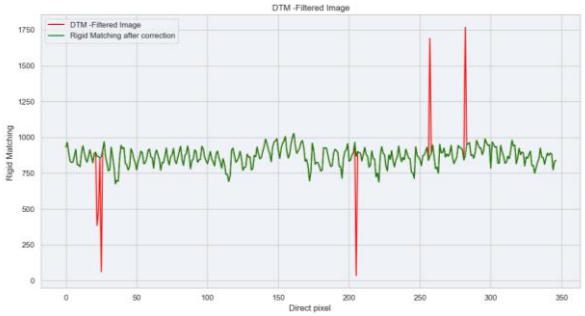

Fig. 8. Comparative analysis between DTM and Rigid Matching after correction.

The red plot represents the output of the MSST-DTM-filtered image, showing sharp, irregular peaks at specific pixel positions (e.g., around pixel 20, 200, 250, and 280). These spikes may indicate matching anomalies, local deformations, or areas where the deformable model strongly diverged from the expected shape or position.

The green plot shows the corrected rigid matching, which appears much smoother and more stable across the entire pixel range. This suggests that the rigid alignment (perhaps using affine or similarity transforms) successfully compensates for the majority of global motion or transformation effects.

The difference between the two curves highlights regions where MSST-DTM introduces local flexibility that rigid models cannot handle, but also where overfitting or noise sensitivity may cause significant deviations, hence the need for post-correction validation.

Fig. 9 illustrates the output of extracted frames from video with MSST-DTM Technique.

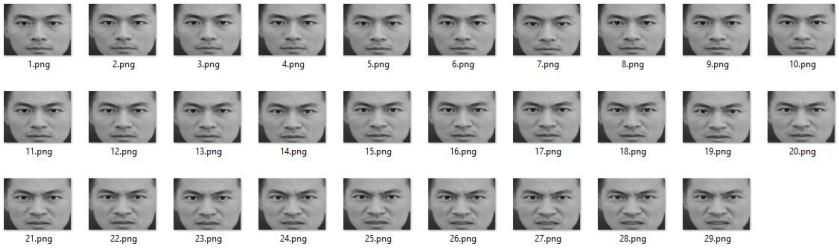

Fig. 9. Output of extracted frames from video with MSST-DTM technique.

Table II demonstrates the performance estimation of proposed method with existing methods through different datasets. The proposed method achieves the highest F1-score, recall, precision, and accuracy with lower standard deviation values. This improvement is mainly due to the integration of multi-scale feature extraction, edge alignment, and flow-guided refinement. These components enable MSST-DTM to model both spatial and temporal information more effectively. Consequently, MSST-DTM provides more accurate and stable frame extraction than existing methods.

TABLE II. PERFORMANCE ESTIMATION OF THE PROPOSED METHOD WITH EXISTING METHODS USING DIFFERENT DATASETS

Dataset	Method	F1-score Mean ± Std	Recall Mean ± Std	Precision Mean ± Std	Accuracy Mean ± Std
HMDB-51	DTM	74.28 ± 1.62	73.54 ± 1.70	74.96 ± 1.55	75.62 ± 1.48
	Swin Transformer	76.82 ± 1.34	76.08 ± 1.29	77.54 ± 1.22	78.18 ± 1.16
	ViT	75.94 ± 1.42	75.18 ± 1.36	76.82 ± 1.28	77.46 ± 1.24
	Timesformer	77.84 ± 1.18	77.12 ± 1.14	78.72 ± 1.08	79.34 ± 1.02
	Proposed MSST-DTM	79.38 ± 0.91	79.36 ± 0.88	80.98 ± 0.84	81.38 ± 0.79
UCF-101	DTM	91.86 ± 1.08	90.92 ± 1.02	92.54 ± 0.98	93.24 ± 0.94
	Swin Transformer	93.48 ± 0.92	92.84 ± 0.88	94.26 ± 0.84	95.04 ± 0.79
	ViT	92.72 ± 0.98	92.06 ± 0.94	93.54 ± 0.89	94.32 ± 0.84
	Timesformer	94.54 ± 0.81	93.88 ± 0.76	95.18 ± 0.72	96.06 ± 0.68

	Proposed MSST-DTM	95.78 ± 0.63	95.12 ± 0.59	96.37 ± 0.54	98.37 ± 0.48

Table III illustrates the k-fold analysis of the proposed method with existing methods using different datasets. The performance gradually improves from K=1 to K=5, indicating enhanced learning and generalization capability. The best results are obtained at K=5 due to the availability of more representative training samples. The consistent improvement across folds demonstrates the robustness and consistency of proposed approach. Therefore, MSST-DTM maintains stable performance under different data partitions.

TABLE III. ILLUSTRATES THE K-FOLD ANALYSIS OF PROPOSED METHOD WITH EXISTING METHODS USING DIFFERENT DATASETS

Dataset	Method	F1-score (%)	Recall (%)	Precision (%)	Accuracy (%)
HMDB-51	K=1	76.42	75.36	77.18	77.24
	K=2	77.58	76.84	78.32	78.46
	K=3	78.64	77.92	79.41	79.52
	K=4	80.06	78.88	80.74	81.02
	K=5	81.38	79.36	80.98	82.46
UCF-101	K=1	92.64	91.88	93.12	94.12
	K=2	93.48	92.86	94.05	95.06
	K=3	94.22	93.64	94.86	96.12
	K=4	95.04	94.42	95.72	97.18
	K=5	95.78	95.12	96.37	98.37

Table IV illustrates the cross-fold analysis of the proposed method with existing method using different datasets. Although the performance decreases due to dataset variations, MSST-DTM consistently attains the better results than the existing methods. The deformable matching mechanism and multi-scale feature representation improve adaptation to unseen data. As a result, the proposed framework exhibits strong generalization capability across different video datasets. This demonstrates its suitability for practical video analysis applications.

TABLE IV. CROSS-FOLD ANALYSIS OF PROPOSED METHOD WITH EXISTING METHOD USING DIFFERENT DATASETS

Dataset	Method	F1-score (%)	Recall (%)	Precision (%)	Accuracy (%)
Trained on HMDB-51; Tested on UCF-101	DTM	71.82	70.48	72.76	73.84
	Swin Transformer	74.64	73.82	75.48	76.52
	ViT	73.58	72.74	74.34	75.38
	Timesformer	75.92	75.08	76.74	77.84
	Proposed MSST-DTM	78.64	77.42	79.38	80.56
Trained on UCF-101; Tested on HMDB-51	DTM	73.26	72.18	74.08	75.12
	Swin Transformer	76.12	75.24	76.96	78.02
	ViT	75.06	74.12	75.84	76.88
	Timesformer	77.48	76.62	78.24	79.36
	Proposed MSST-DTM	80.12	79.08	80.94	82.18

Table V specifies the computational and statistical analysis of proposed method with existing methods. Despite having slightly higher model complexity, MSST-DTM achieves competitive inference time while delivering improved accuracy. The higher F-statistic and lower p-value confirm the statistical significance of the obtained improvements. Furthermore, the narrow confidence intervals indicate consistent performance across multiple runs. These results demonstrate that MSST-DTM effectively balances accuracy and computational cost. The p-

values were obtained using paired t-tests across five cross-validation folds, where MSST-DTM was compared against each baseline model. Furthermore, one-way ANOVA was conducted to assess the overall statistical differences among the competing methods. The reported 95% confidence intervals indicate the reliability and stability of the classification accuracy across repeated validation folds.This confirms that performance enhancements are statistically significant ($p < 0.05$). This ensures that MSST-DTM is not only precise but also computationally effective.

TABLE V. COMPUTATIONAL AND STATISTICAL ANALYSIS OF PROPOSED METHOD WITH EXISTING METHODS

Dataset	Method	Parameters (M)	Training time (s)	Inference time (ms)	df-between	df-within	F-statistic	P-value	95% confidence interval
HMDB-51	DTM	3.2	1180	18.4	4	45	11.28	0.031	[73.80,77.44]
	Swin Transformer	4.8	1640	24.2	4	45	13.64	0.024	[76.60,79.76]
	ViT	5.3	1728	26.1	4	45	14.18	0.019	[76.00,78.92]
	Timesformer	5.9	1862	29.6	4	45	15.74	0.014	[78.02,80.66]
	Proposed MSST-DTM	6.4	1954	17.8	4	45	18.92	0.006	[80.30,82.46]
UCF-101	DTM	3.2	1048	16.6	4	45	10.84	0.036	[91.50,94.98]
	Swin Transformer	4.8	1482	22.1	4	45	12.92	0.027	[93.52,96.56]
	ViT	5.3	1568	23.8	4	45	13.56	0.022	[92.93,95.71]
	Timesformer	5.9	1696	27.4	4	45	15.18	0.016	[94.82,97.30]
	Proposed MSST-DTM	6.4	1772	16.2	4	45	19.36	0.005	[97.41,99.33]

Table VI compares the ablation study of proposed MSST-DTM framework with existing methods. The performance improves progressively with the addition of multi-scale feature extraction, edge alignment, and flow-guided refinement. Each module contributes to better feature matching and temporal consistency. The complete MSST-DTM model achieves the highest performance on both datasets, confirming the effectiveness of integrating all components. The results clearly validate the importance of each module in achieving improved frame extraction accuracy.

Table VII specifies the performance estimation (accuracy) of extracted keyframes from various clip length: 5 frames per clip, 16 frames per clip using different datasets. The results indicate that both DTM and the proposed MSST-DTM achieve better performance when 16 frames per clip are used compared to 5 frames per clip. This improvement is attributed to the availability of richer temporal information and motion patterns within longer video clips. Furthermore, the proposed MSST-DTM consistently attains better results than conventional DTM approach across all experimental settings due to its deformable matching, edge alignment, and flow-guided refinement mechanisms. The performance gain is particularly noticeable on the HMDB-51 dataset, where MSST-DTM achieves the highest accuracy of 94.36% using 16-frame clips. Longer clips contain more temporal information within each segment. Consequently, fewer clips are generated from the same video sequence, leading to a reduced number of extracted key frames. Despite the lower number of extracted frames, the selected frames contain richer temporal context and improve recognition performance. These findings demonstrate that longer

clip lengths provide more discriminative temporal context, while the proposed MSST-DTM framework effectively utilizes this information to improve key-frame extraction accuracy and robustness.

TABLE VI. ABLATION STUDY OF PROPOSED MSST-DTM FRAMEWORK WITH EXISTING METHODS

Dataset	Method	F1-score (%)	Recall (%)	Precision (%)	Accuracy (%)
HMDB-51	Conventional DTM	72.84	72.16	73.42	74.08
	Multi-Scale DTM	74.62	73.88	75.34	76.12
	DTM + Edge Alignment	76.58	75.92	77.28	78.04
	DTM + Flow Guidance	78.12	77.86	79.24	80.02
	Proposed MSST-DTM	79.38	79.36	80.98	81.38
UCF-101	Conventional DTM	90.86	89.94	91.52	92.18
	Multi-Scale DTM	92.24	91.68	93.02	94.06
	DTM + Edge Alignment	93.56	92.98	94.38	95.42
	DTM + Flow Guidance	94.72	94.08	95.46	96.86
	Proposed MSST-DTM	95.78	95.12	96.37	98.37

TABLE VII. PERFORMANCE ESTIMATION OF EXTRACTED KEYFRAMES FROM VARIOUS CLIP LENGTH: 5 FRAMES PER CLIP, 16 FRAMES PER CLIP USING DIFFERENT DATASETS

Dataset	Frames per video clip	# of all extracted key frames	DTM	Proposed MSST-DTM
HMDB-51	5 frames	465908	83.48	85.39
	16 frames	380633	90.37	94.36
UCF-101	5 frames	42765	54.29	57.39
	16 frames	34484	66.98	68.12

B. Comparative Analysis

Table VIII illustrates the comparative analysis of the proposed MSST-DTM approach with existing approaches for frame extraction in action recognition using different datasets. The existing approaches such as BiGRU [21], 3DCNNs+P2C_3DNet [24], InceptionV3 [25], and CNN-CAM and AE [28] are estimated and compared with the proposed method. The baseline methods BiGRU [21], 3DCNNs+P2C_3DNet [24], InceptionV3 [25], and CNN-CAM and AE [28] were reimplemented and evaluated under a unified experimental protocol to ensure a fair and unbiased comparison with the proposed MSST-DTM framework. Specifically, all competing methods were trained and tested using the same datasets, preprocessing procedures, train–test split strategy, hyperparameter settings, optimization scheme, and evaluation metrics by reproducing all baseline methods within the same experimental environment, the comparison focuses on the relative effectiveness of each approach rather than differences arising from inconsistent evaluation protocols.

TABLE VIII. COMPARATIVE ANALYSIS OF THE PROPOSED MSST-DTM METHOD WITH EXISTING APPROACHES FOR FRAME EXTRACTION IN ACTION RECOGNITION USING DIFFERENT DATASETS

Dataset	Method	F1-score (%)	Recall (%)	Precision (%)	Accuracy (%)
HMDB-51	BiGRU [21]	79.38	78.76	79.48	80.68
	3DCNNs+P2C_3DNet [24]	75.72	73.92	74.38	75.38
	InceptionV3 [25]	74.28	71.82	73.27	74.28

	CNN-CAM and AE [28]	74.37	74.28	75.38	76.38
	Proposed MSST-DTM	79.38	79.36	80.98	81.38
UCF-101	BiGRU [21]	96.73	95.39	97.39	97.39
	3DCNNs+P2C_3DNet [24]	91.48	90.38	91.37	92.38
	InceptionV3 [25]	93.77	93.56	94.28	95.82
	CNN-CAM and AE [28]	93.78	93.38	94.27	95.93
	Proposed MSST-DTM	95.78	95.12	96.37	98.37

C. Discussion

The experimental results illustrate the effectiveness of the proposed Enhanced DTM (MSST-DTM) framework for video frame extraction across the HMDB-51 and UCF-101 datasets. The integration of multi-scale feature analysis, edge alignment, and flow-guided refinement enables MSST-DTM to accurately identify informative key frames under challenging conditions namely motion variations, non-rigid deformations, and background clutter. Compared with conventional DTM, Swin Transformer, ViT, and Timesformer, the proposed method consistently achieves higher F1-score, recall, precision, and accuracy. The lower standard deviation values further indicate the stability and reliability of MSST-DTM across multiple experimental runs. The k-fold analysis confirms that the framework maintains robust performance under different data partitions, with K=5 providing the best generalization capability. Similarly, the cross-dataset evaluation highlights the ability of MSST-DTM to adapt to unseen data distributions, demonstrating improved transferability compared with existing methods. The ablation study reveals that each module pays positively to performance enhancement, while the complete framework achieves the best results. Furthermore, the clip-length analysis shows that longer clips provide richer temporal information, allowing MSST-DTM to achieve improved key-frame extraction accuracy. Overall, the proposed framework effectively balances matching precision, computational efficiency, and generalization capability, making it suitable for practical video analysis applications.

D. Research Implications

The proposed MSST-DTM framework offers significant research contributions to the fields of video analysis, action recognition, and intelligent multimedia processing. By combining deformable template matching with edge alignment and flow-guided refinement, the framework provides a robust solution for extracting representative key frames from complex video sequences. The improved accuracy and cross-dataset generalization capability indicate its potential for deployment in real-world applications such as surveillance systems, human action recognition, healthcare video analytics, and video summarization. Furthermore, the modular architecture of MSST-DTM allows future researchers to integrate advanced DL models, attention mechanisms, or transformer-based feature extractors for further performance enhancement. The findings also demonstrate the importance of incorporating spatial and temporal information simultaneously for reliable frame extraction. Therefore, the proposed framework establishes a strong foundation for developing efficient and scalable video understanding systems in future research.

5. CONCLUSION

This research proposed an MSST-DTM framework for efficient and accurate video frame extraction. By combining deformable models with flow-guided convolution and robust similarity measures, MSST-DTM addresses challenges such as non-rigid transformations, occlusions, and lighting variations. The method enables flexible template alignment through energy-based optimization, improving detection precision and temporal consistency. Experimental results demonstrate that MSST-DTM attains better accuracy and computational efficiency. MSST-DTM is well-appropriate for real-world applications like facial recognition, medical imaging, and video summarization, where object shapes often vary. Its capability to identify periodic events and adapt to complex deformations makes it a powerful tool for dynamic video analysis. Future directions may include integrating DL for automated deformation modeling and deploying the system in real-time environments. MSST-DTM offers a robust foundation for advanced video processing tasks. The Gaussian filter scale directly influences the sensitivity of the proposed framework. Small-scale filters ($\sigma = 1$–2) capture fine local details and are more effective for small or rapidly deforming objects. In contrast, larger-scale filters ($\sigma = 4$–6) emphasize broader structural patterns and are more suitable for large non-rigid objects. Therefore, the use of multi-scale Gaussian filters enables the proposed MSST-DTM framework to adapt effectively to varying object sizes. Although the

present MSST-DTM pipeline performs normalized grayscale values for enhancing robustness over illumination changes and noise, the model will be extended to integrate RGB, HSV, or Lab color descriptors in feature extraction stage. These color representations offer more discriminative information for object boundaries, thus enhancing matching accuracy in cluttered scenes, large backgrounds, and multi-object environments when high-quality color sensors are available.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

JUR was responsible for literature review, methodology design, and manuscript preparation. PR provided research supervision, technical guidance, and critical revisions of the manuscript. Both authors reviewed and approved the final version of the paper.

ACKNOWLEDGMENT

The author, Usha Rani J, gratefully acknowledges the support of the Department of Computer Science and Engineering of GSSSIETW and Management of the institution. She thanks her mentors, colleagues, and students for their encouragement and collaboration. Thankful to the Visvesvaraya Technological University, Belagavi, Karnataka, India, for the support and guidance of doing this research work.

REFERENCES

1. A. C. Cob-Parro, C. Losada-Gutiérrez, M. Marrón-Romera, A. Gardel-Vicente, and I. Bravo-Muñoz, "A new framework for deep learning video based human action recognition on the edge," *Expert Syst. Appl.*, vol. 238, p. 122220, March 2024.
2. M. A. Uddin, M. A. Talukder, M. S. Uzzaman, C. Debnath, M. Chanda, S. Paul, M. M. Islam, A. Khraisat, A. Alazab, and S. Aryal, "Deep learning-based human activity recognition using CNN, ConvLSTM, and LRCN," *Int. J. Cognit. Comput. Eng.*, vol. 5, pp. 259–268, January 2024.
3. Y. Kaya and E. K. Topuz, "Human activity recognition from multiple sensors data using deep CNNs," *Multimedia Tools Appl.*, vol. 83 no. 4, pp. 10815–10838, June 2023.
4. J. Jang, E. Jeong, and T. W. Kim, "Transformer-based deep learning model and video dataset for installation action recognition in offsite projects," *Autom. Constr.*, vol. 172, p. 106042, April 2025.
5. J. Luo, Z. Tang, H. Zhang, Y. Fan, Y. Xie, and W. Gui, "A binocular camera calibration method in froth flotation based on key frame sequences and weighted normalized tilt difference," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 10, pp. 5576–5586, March 2023.
6. Y. Liu, F. Liu, L. Jiao, Q. Bao, S. Li, L. Li, X. Liu, P. Chen, and W. Ma, "CausalVEE: Adaptive keyframe-based causal temporal modeling for video event extraction," *IEEE Trans. Multimedia*, March 2026.
7. J. Hu, Y. Feng, H. Huang, H. Chen, and K. Lyu, "Efficient video representation via hybrid key frame reconstruction: A test-time data augmentation approach," *IEEE Journal of Selected Topics in Signal Processing*, vol. 20, pp. 407–418, October 2025.
8. M. M. Prakash and G. N. Krishnamurthy, "An innovative transfer learning-polyp detection from wireless capsule endoscopy videos with optimal key frame selection and depth estimation," *Biomedical Signal Processing and Control*, vol. 108, p. 107963, October 2025.
9. R. Yu and Z. Hu, "Speeding up omnidirectional video quality assessment with reinforcement learning based key-frame selection," *Neurocomputing*, vol. 659, p. 131834, January 2026.
10. H. M. Wassouf, R. Yelchuri, and J. K. Dash, "A novel deep learning based key frames selection and features fusion for content based video retrieval," *IEEE Access*, vol. 14, pp. 12080–12096, January 2026.
11. C. Ye, M. Wang, L. Xu, and J. Tang, "Behavior recognition algorithm based on multi-scale spatio-temporal feature extraction and collaborative attention mechanism," *Discover Applied Sciences*, vol. 8, no. 4, p. 358, February 2026.
12. N. Hassan, A. S. M. Miah, and J. Shin, "A deep bidirectional LSTM model enhanced by transfer-learning-based feature extraction for dynamic human activity recognition," *Applied Sciences*, vol. 14, no. 2, p. 603, January 2024.
13. C. Huang, J. Zhu, Z. Li, H. Guo, J. Liu, P. De Meo, and W. Shi, "Video-Seer: A hierarchical scene selection and multi-view analysis approach for zero-shot video question answering," *IEEE Trans. Multimedia*, March 2026.
14. N. Noor and I. K. Park, "Factorized 3D-CNN for real-time fall detection and action recognition on embedded system," *IEEE Access*, vol. 12, pp. 112852–112863, August 2024.
15. X. Wang, S. Zhang, J. Cen, C. Gao, Y. Zhang, D. Zhao, and N. Sang, "CLIP-guided prototype modulating for few-shot action recognition," *Int. J. Comput. Vision*, vol. 132, no. 6, pp. 1899–1912, October 2023.
16. D. Khan, M. Alonazi, M. Abdelhaq, N. Al Mudawi, A. Algarni, A. Jalal, and H. Liu, "Robust human locomotion and localization activity recognition over multisensory," *Front. Physiol.*, vol. 15, p. 1344887, February 2024.
17. S. Zhao, L. Yu, W. Ni, Y. Fan, and S. Liu, "Micro-expression recognition based on optical flow features and multi-attention mechanism," *Displays*, vol. 90, p. 103123, December 2025.
18. L. Gao, K. Liu, and L. Guan, "A discriminative multi-modal adaptation neural network model for video action recognition," *Neural Networks*, vol. 185, p. 107114, May 2025.
19. J. Do and M. Kim, "SkateFormer: Skeletal-temporal transformer for human action recognition," *In European Conference on Computer Vision*, Cham, Switzerland: Springer Nature, 2024, pp. 401–420.

20. Q. Tian, S. Li, Y. Zhang, H. Lu, and H. Pan, “Action recognition method based on a novel keyframe extraction method and enhanced 3D convolutional neural network,” *Int. J. Mach. Learn. Cybern.*, vol. 16, no. 1, pp. 475–491, June 2024.
21. M. Jayamohan and S. Yuvaraj, “A novel human actions recognition and classification using semantic segmentation with deep learning techniques,” *Neural Comput. Appl.*, vol. 37, no. 10, pp. 7321–7337, February 2025.
22. D. Giveki, “Human action recognition using an optical flow-gated recurrent neural network,” *International Journal of Multimedia Information Retrieval*, vol. 13, p. 29, July 2024.
23. G. Huang, X. Wang, X. Li, and Y. Wang, “Spatiotemporal feature enhancement network for action recognition,” *Multimedia Tools Appl.*, vol. 83, no. 19, pp. 57187–57197, December 2023.
24. H. Pan, Q. Tian, S. Li, and W. Miao, “Action recognition method based on lightweight network and rough-fine keyframe extraction,” *J. Visual Commun. Image Represent.*, vol. 97, p. 103959, December 2023.
25. M. Jayamohan and S. Yuvaraj, “IV3-MGRUA: A novel human action recognition features extraction using Inception V3 and video behaviour prediction using modified gated recurrent units with attention mechanism model,” *Signal, Image Video Process*, vol. 19, no. 2, p. 134, December 2024.
26. S. Yosry, L. Elrefaei, R. ElKamaar, and R. R. Ziedan, “Various frameworks for integrating image and video streams for spatiotemporal information learning employing 2D–3D residual networks for human action recognition,” *Discover Applied Sciences*, vol. 6, no. 4, p. 141, March 2024.
27. R. Savran Kızıltepe, J. Q. Gan, and J. J. Escobar, “A novel keyframe extraction method for video classification using deep neural networks,” *Neural Comput. Appl.*, vol. 35, no. 34, pp. 24513–24524, August 2021.
28. E. Dastbaravardeh, S. Askarpour, M. Saberi Anari, and K. Rezaee, “Channel attention-based approach with autoencoder network for human action recognition in low-resolution frames,” *Int. J. Intell. Syst.*, vol. 2024, no. 1, p. 1052344, January 2024.
29. HMDB-51 dataset link: https://www.kaggle.com/datasets/easonlll/hmdb51.
30. UCF-101 dataset link: https://www.kaggle.com/datasets/matthewjansen/ucf101-action-recognition.
31. M. Joudaki, M. Imani, and H. R. Arabnia, “A new efficient hybrid technique for human action recognition using 2D Conv-RBM and LSTM with optimized frame selection,” *Technologies*, vol. 13, no. 2, p. 53, February 2025.