Ideas from Literature Review

7 May 2025

Extracting more features by post-processing

- Flow vectors

  Movement of keypoints (body joints) between consecutive frames, captures direction and speed of motion for each joint.

- Angles Between Anchor Joints

  Angles between sets of three body joints that form triangles. The paper does not list a fixed set of joint triplets explicitly.

- Normalized Distance Vectors

  Distances between key joints normalized to remove scale differences

- SIFT (Scale-Invariant Feature Transform)
  - Extract robust local keypoint descriptors from the image that are invariant to scale, rotation, and illumination. Helps maintain consistent keypoint detection even when the camera moves or zooms in/out.
  - In this paper SIFT is applied to the grayscale version of each frame (Rani & Devarakonda, 2022).
- LoG (Laplacian of Gaussian)
  - Rani and Devarakonda (2022) applied LoG to a distance transform image (DT), where background pixels represent their distance from the nearest foreground joint.
  - LoG helps cluster pixels around joints.
  - Only stable maxima are kept → more accurate joint center estimates.
- Cosine Similarity
  - The angular similarity between two poses: the student's pose vector and the tutor's pose vector.

CNN-LSTM (Classification)

From Rani and Devarakonda (2022):

CNN Block:

- Conv layer C-1: 3×3 filters + ReLU.
- Conv layers C-2 and C-3 (similar structure).
- Pooling: 2×2 kernel (P-1).
- Output: Flatten + Dense (512 units).

LSTM Block (Standard (vanilla) LSTM):

- Receives flattened CNN features as input.
- Transforms to time-sequence data.
- 54-neuron Dense layer follows LSTM.

Classifier: Dense layer followed by 25% Dropout and softmax for final classification into 8 dance classes.

Training Details:

- 65 Eochs
- Batch size: 32
- input size: (48 frames, 2048 features)
- Didn’t state Optimizer and Loss Function

LSTM-Based Classification Model (Sequence Learning)

From Srivastava, Umrao, and Yadav (2024):

Sequential Model:

- LSTM layer 1: 64 units, return_sequences=True, activation='relu'
- LSTM layer 2: 128 units, return_sequences=True, activation='relu'
- LSTM layer 3: 64 units, activation='relu'
- Dense layer 1: 64 units, activation='relu'
- Dense layer 2: 32 units, activation='relu'
- Output layer: Dense, 10 units (for 10 yoga pose classes), activation='softmax'

Training Details:

- Input shape: (30, 1662) → 30 time steps, 1662 features per frame
- **Optimizer**: Adam; **Loss**: Categorical Cross Entropy; **Epochs**: 100; **Batch size:** 16

References

Liu, J., Shahroudy, A., Perez, M., Wang, G., Duan, L. Y., & Kot, A. C. (2019). Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. *IEEE transactions on pattern analysis and machine intelligence*, *42*(10), 2684-2701.

Rani, C. J., & Devarakonda, N. (2022). An effectual classical dance pose estimation and classification system employing convolution neural network–long short term memory (CNN-LSTM) network for video sequences. *Microprocessors and Microsystems*, *95*, 104651.

Srivastava, R. P., Umrao, L. S., & Yadav, R. S. (2024). Real-time yoga pose classification with 3-D Pose Estimation Model with LSTM. *Multimedia Tools and Applications*, *83*(11), 33019-33030.