# Key Notes, Key Features: A Temporal Convolutional Approach to Pianist Technique Classification

Max Rodriguez* & Renn Su*

CS 131 Winter 2025



Stanford University

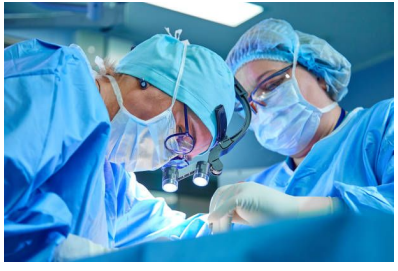
# Introduction

**Outline:**

1. Introduction/Background
2. Data featurization and pre-processing
3. Activity Clustering
4. CNN Model Architecture/Proficiency level classification
5. Results
6. Discussion
7. Questions

# Movement at scale—smaller movements are harder to understand

Gross-Motor Skills



Fine-Motor Skills

# What are the features that make up a fine-motor task?

Our case study: piano playing

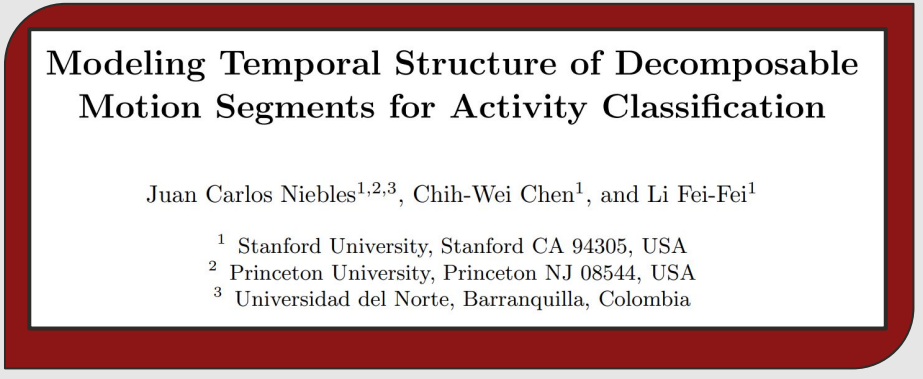
- Technique
- Wrist posture
- Finger dexterity
- Accuracy of motion
- Musicality and rhythm
- etc.

Can we use CV tools/methods to capture these features and use them to classify pianist fine-motor skill level?

# Previously in computer vision…

David Johnson, Daniela Damian, and George Tzanetakis
Department of Computer Science
University of Victoria
3800 Finnerty Road
Engineering and Computer Science
Building, Room 504
Victoria, BC V8P 5C2 Canada
davidjo@uvic.ca, danielad@uvic.ca,
gtzan@ieee.org

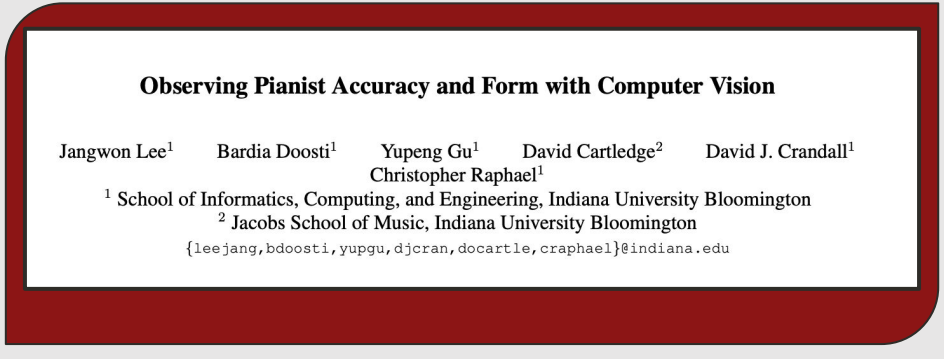
**Detecting Hand Posture in Piano Playing Using Depth Data**



Modeling Temporal Structure of Decomposable Motion Segments for Activity Classification

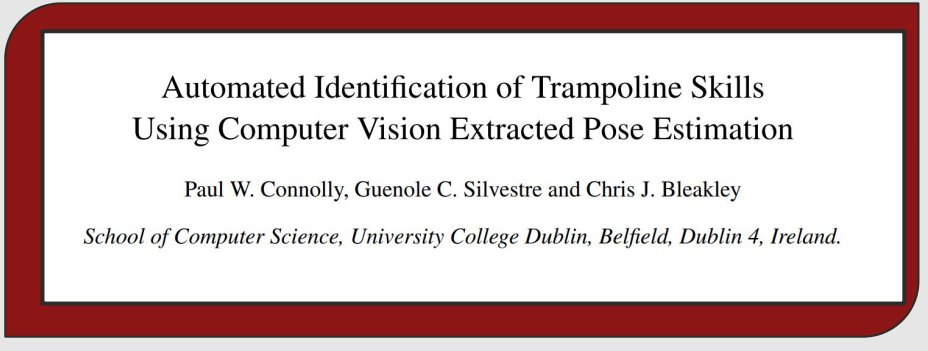
Juan Carlos Niebles[1,2,3], Chih-Wei Chen[1], and Li Fei-Fei[1]

[1] Stanford University, Stanford CA 94305, USA
[2] Princeton University, Princeton NJ 08544, USA
[3] Universidad del Norte, Barranquilla, Colombia



**Observing Pianist Accuracy and Form with Computer Vision**

Jangwon Lee[1] Bardia Doosti[1] Yupeng Gu[1] David Cartledge[2] David J. Crandall[1]
Christopher Raphael[1]
[1] School of Informatics, Computing, and Engineering, Indiana University Bloomington
[2] Jacobs School of Music, Indiana University Bloomington
{leejang,bdoosti,yupgu,djcran,docartle,craphael}@indiana.edu



Automated Identification of Trampoline Skills Using Computer Vision Extracted Pose Estimation

Paul W. Connolly, Guenole C. Silvestre and Chris J. Bleakley

*School of Computer Science, University College Dublin, Belfield, Dublin 4, Ireland.*

Stanford University

# Data Featurization and Preprocessing

# Data Collection

**Data collected in-person and off YouTube from:**

- 50 beginner-level pianists
- 50 advanced-level pianists

**Multiple Motion Sequences:**

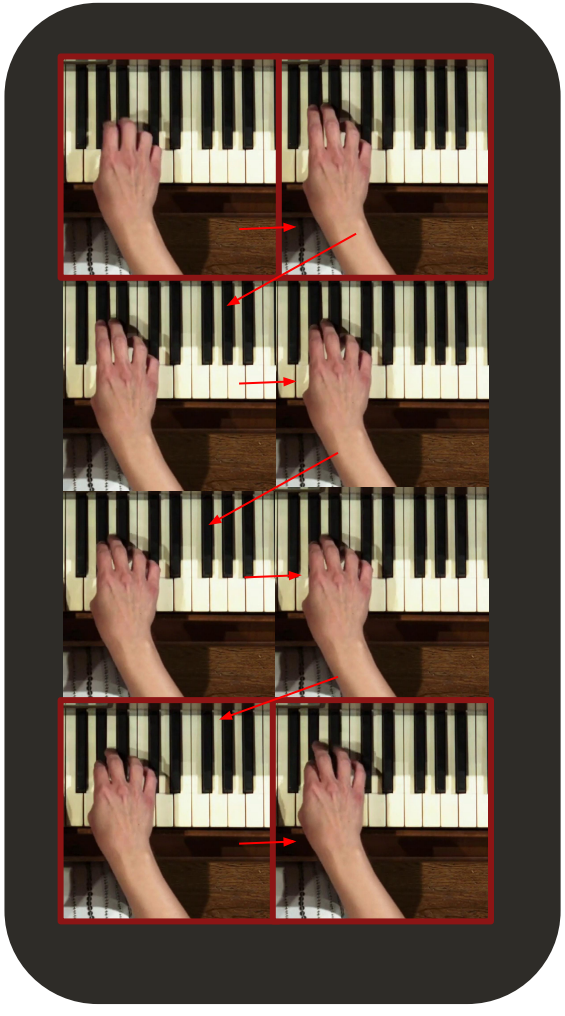
- Scales
- Arpeggios
- Chord progressions, etc.

**Video Collection Setup:**

- Bird's-eye view of the back of the hand
- Both left and right hands isolated via further video processing

**Data Extraction:**

- Each video yielded 10 sequences (separated by 1 frame)
- Each sequence consists of 20 images



# Rotation and Scaling

- Google's **MediaPipe** detects **21** key landmarks per hand image.
- **Hand Orientation Normalization:** Rotated using angle between a unit y-axis vector and the wrist-to-middle finger MCP vector:

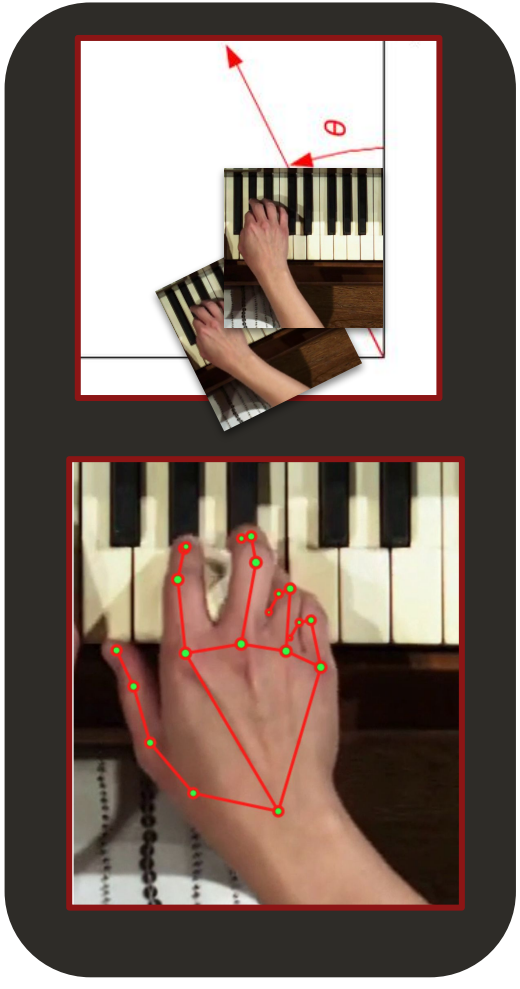
$$\mathbf{v} = \text{MCP}_{kp} - \text{Wrist}_{kp}$$

$$\theta = \text{degrees}\left(\arctan 2(v_y, v_x)\right)$$

- **Feature Scaling:** Keypoints, greyscale values, and optical flow standardized by subtracting mean and dividing by std across the dataset.

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad \sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2},$$

$$X_{\text{normalized}} = \frac{X - \mu}{\sigma}.$$



# Key-landmark Detection and Optical Flow

**Harris Corner Detection:**

- Computes keypoints via corner response:

$$R = Det(M) - k(Trace(M)^2) \text{ at each pixel}$$

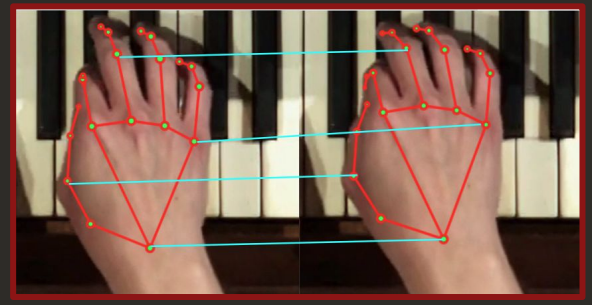
**SIFT:**

- Uses a 16×16 neighborhood to build descriptors from gradient magnitudes and orientations
- Keypoint matching: accepts a match if distance to 1st match << distance to 2nd match
- Filters out unstable keypoints (high variance or out-of-frame)

**Lucas-Kanade Optical Flow:**

- Calculates displacement for keypoints across frames
- Overdetermined system solved iteratively by minimizing:

$$\min_{\Delta u,\, \Delta v} \sum \left(I_x \Delta u + I_y \Delta v + I_t\right)^2$$



$$A = \begin{bmatrix} I_x(q_1) & I_y(q_1) \\ I_x(q_2) & I_y(q_2) \\ \vdots & \vdots \\ I_x(q_n) & I_y(q_n) \end{bmatrix} \quad v = \begin{bmatrix} V_x \\ V_y \end{bmatrix} \quad b = \begin{bmatrix} -I_t(q_1) \\ -I_t(q_2) \\ \vdots \\ -I_t(q_n) \end{bmatrix}$$

$$\mathrm{v} = (A^T A)^{-1} A^T b$$

# Preprocessing for Classification

**Keypoint Extraction:**

- For each image row, top 10 keypoints selected based on highest optical flow displacement (most activity).
  - Optical flow magnitude computed as: $\text{magnitude} = \sqrt{v_x^2 + v_y^2}$
- Each keypoint represented by 5 values (flattened into one row):
  - x coordinate, y coordinate, grayscale value, optical flow x value, optical flow y value

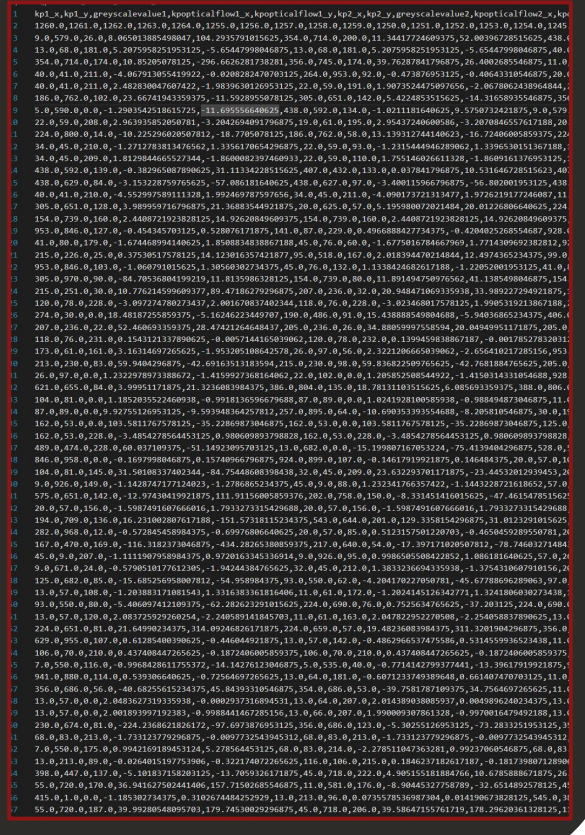
**Sequence Formation:**

- 15 rows per sequence constitute one data point.
- Each sequence is labeled as:
  - 1 for advanced-level pianists
  - 0 for beginner-level pianists

**Dataset Composition:**

- Holdout Dataset:
  - 100 beginner sequences
  - 100 advanced sequences
- Training Dataset:
  - 200 beginner sequences
  - 200 advanced sequences

**Advanced Holdout Data for keypoints with top optical flow…**



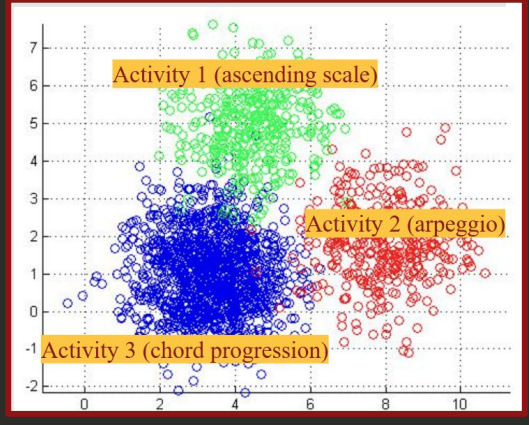
# K-means activity Clustering

**Challenge:**

- Diverse pianist motions (e.g., sloppy beginner scales vs. advanced arpeggios) can confuse a proficiency-level CNN classifier.

**Solution:**

- Cluster Training Data:
  - Apply k-means hierarchical clustering to group sequences by similar motion.
  - Used 3 clusters (due to small training set size).
- Holdout Assignment:
  - Match each holdout sequence to closest cluster centroid.
  - Evaluate using CNN trained on corresponding activity cluster training data.



Stanford University

# Convolutional Neural Network Architecture

# How do we account for ***both*** single fingers and the whole hand?

# Motivating CNN Architectures

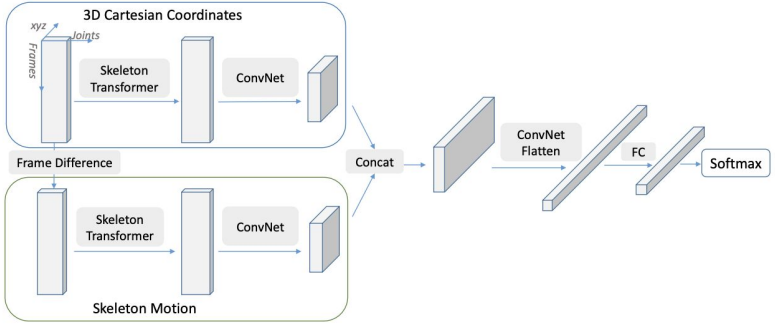



**Fig. 1**. CNN representation of skeleton sequences for action classification.

Multi-stream CNN for skeletal keypoint classification (Li et al.)

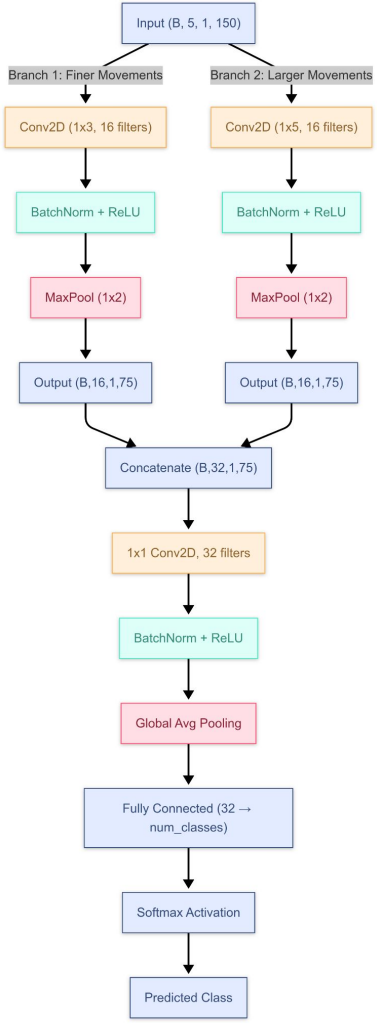
**Network In Network**

**Min Lin[1,2], Qiang Chen[2], Shuicheng Yan[2]**
[1]Graduate School for Integrative Sciences and Engineering
[2]Department of Electronic & Computer Engineering
National University of Singapore, Singapore
{linmin,chenqiang,eleyans}@nus.edu.sg

1x1 Convolutions for dimensionality reduction and feature fusions (Lin et al.)

# Our CNN Architecture

- **Two branches:** local movements and global movements
  - i.e. finger-level (1x3 kernel) and hand-level (1x5 kernel) features
- **Maxpool:** highlighting dominant motion patterns
  - 150 ⟶ 75 input width
- **Aggregation of two branches:** concatenation and feature fusion steps identify interactions between fine & broad movements



# CNN Performance

- The two-branch architecture outperformed single branch by **20.9%**
- Though clustering reduced available data, performance was comparable
- Fewer advanced players were incorrectly classified as beginners (9/99, 11%)
  - "Beginner" covered a much wider range of players

Table 1. Comparison of model performance on training/validation and holdout data.

| Model | Training/Validation | Holdout Data |
|---|---|---|
| Baseline Using MLP | 60.3% | 53.9% |
| Single Branch CNN | 100.0% | 54.1% |
| Two-Branch CNN | 98.7% | **75.0%** |
| Two-Branch CNN with Clustering | 100.0% | 72.7% |

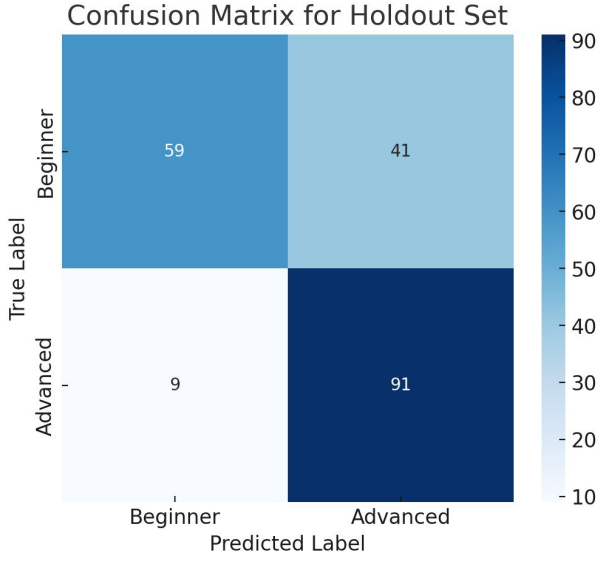



Table 2. Confusion Matrix for Two-Branch CNN

# Significance of Work + Future Directions

Our architecture achieved **75%** accuracy in skill classification for piano—a fine motor task.

- The number is comparable to fine motor skill classification in activities such as medical surgery. 75% (Kitaguchi et al., 2021) and 87% (Lavanchy et al., 2021)
  - However, we achieved these results using significantly less data (e.g. 400 sequences vs 1480 sequences) and weaker hardware.
- More data for more activity clusters, potential intermediate class

Stanford University

# Any questions?

Thank you for listening!