

# Action Recognition

Topics in Computer Science 1 (CSCI 4440U)

**Faisal Z. Qureshi**

<http://vclab.science.ontariotechu.ca>



# A bit about me



## **Faisal Qureshi**

Professor

Computer Science

Visual Computing Lab

Faculty of Science

Ontario Tech University (formerly UOIT)

✍ UA4000, 2000 Simcoe St. N., Oshawa, ON L1G 0C5 Canada

✉ faisal.qureshi@uoit.ca

☎ (905) 721-8668 x 3626

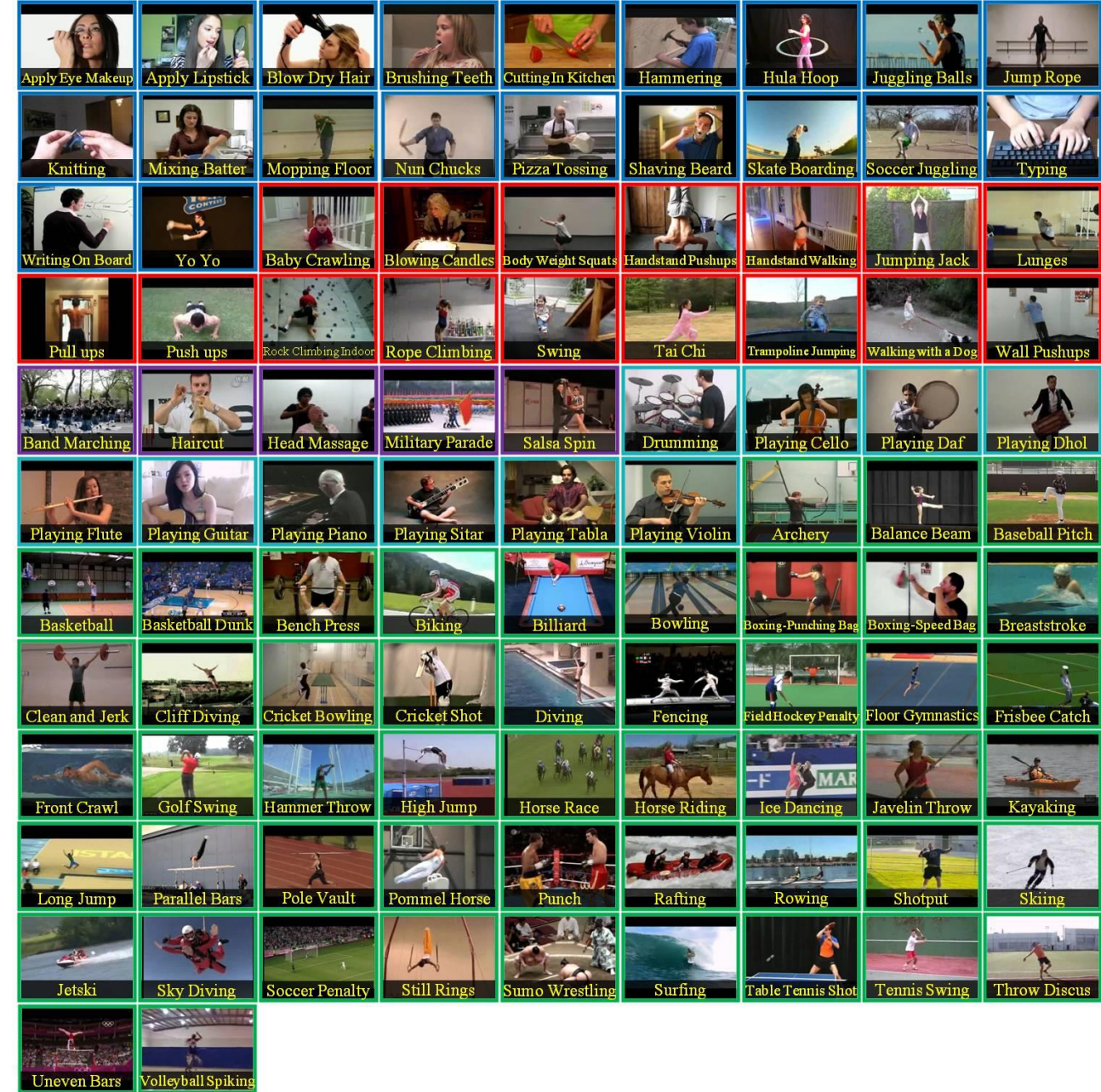
<http://www.vclab.ca>

# Important questions

- Will I get an A+ in this course?
- Why should I care about action recognition?

# Action recognition

- Action recognition is the task of identifying specific human actions or activities within video sequences or static images.



<https://www.crcv.ucf.edu/research/data-sets/ucf101/>

# Why?

- Video understanding
- Video indexing and retrieval
- Content navigation

# Applications

- Surveillance
- Human-computer interaction
- Sports analysis
- Healthcare
- Entertainment

# Beyond applications

- Key component in advancing AI-driven applications
  - Autonomous vehicles
  - Smart homes
  - Advanced robotics

# This course

- Not a typical undergraduate courses
- You'll be asked to read papers, critique them, and implement one to two papers
- The midterm exams will cover background material (plus the papers that we discuss in the class)
- It is best if you have some background in computational photography, computer vision, and machine learning.
  - Comfort with linear algebra and calculus is also necessary



# Course website

- <https://csundergrad.science.uoit.ca/courses/csci4440u-f24/>



# Topics (loosely speaking)

- Fundamentals of action recognition
- Feature extraction and representation
- Machine learning and deep learning
- Spatio-temporal modeling
- Datasets and evaluation
- Applications and case studies
- Emerging trends

# Learning outcomes

- Understand and implement key algorithms and models used for action recognition in images and videos.
- Analyze and critique the strengths and weaknesses of various action recognition approaches.
- Apply advanced machine learning and deep learning techniques to solve real-world action recognition problems.
- Develop and evaluate action recognition systems using state-of-the-art tools and datasets.
- Stay abreast of the latest research and trends in the rapidly evolving field of action recognition.

# Grading

- In-class discussions 20%
- Midterm exams 40%
- Course project 40%

*A student must get 50% in the course project to pass the course.  
Furthermore, a student must get 50% in the two midterms to pass the course.  
Class attendance is **not** optional.*

# Papers

- Each week a paper will be assigned to one or more students, who will lead the discussion on that paper
- Paper list will be made available on the course website

# Course project

- The course project is an independent exploration of a specific problem within the context of this course.
- Project will consist of implementing one or more papers
- Project grade will depend on the ideas, how well you present them in the report, how well you position your work in the related literature, how thorough are your experiments and how thoughtful are your conclusions.
- *Teams of up to two students are allowed.*
- *You are required to prepare a three-minutes video that provides an overview of your project.*
- You are also required to prepare a final project writeup.

# Important dates

- Midterm 1 on Oct 4
- Study break during the week of Oct 14
- Midterm 2 on Nov 18
- Project selection due by Oct 18
  - *You may lose up to 10% of the course project grade if project selection isn't finalized by Oct 18. You may lose up to an additional 20% of the course project grade if the project selection isn't finalized by Oct 25.*
- Project report due on Dec 8, by 11:59 pm
  - *You may be asked to record a 3 minutes long project presentation that will be submitted before the last week of lectures.*

# Course syllabus

- Please consult course syllabus available at the course website for policies regarding conduct, late submissions, remarking, etc.



# Questions?

# Early methods

- Template matching
  - Comparing video frames to pre-defined templates
- Statistical methods
  - Using Hidden Markov Models (HMM) and Gaussian Mixture Models (GMM) to model action sequences

# Machine learning

- Using Support Vector Machines (SVM) and Random Forests to classify actions based on extracted features

# Deep learning

- Convolutional neural networks
- Recurrent neural networks
- Transformers
  
- End-to-end action recognition

# Challenges

- Intra-class Variability
  - Actions can be performed in different styles by different people (e.g., different ways of walking or running).
  - Variability due to speed, viewpoint, and individual differences.

# Challenges

- Inter-class Similarity
  - Some actions may look very similar but belong to different classes (e.g., walking and jogging).
  - Fine-grained differences that are hard to capture.

# Challenges

- Background Clutter
  - Complex or dynamic backgrounds that make it difficult to isolate the action.
  - Need for robust background subtraction or segmentation techniques.

# Challenges

- Temporal Dynamics
  - Actions unfold over time, requiring models to capture temporal sequences accurately.
  - Temporal dependencies and the need for models to recognize the correct order of frames.



# Challenges

- Occlusions and Partial Observations
  - Parts of the action may be hidden or only partially visible, requiring robust feature extraction and inference.

# Spatial features

- Extracted from individual frames of the video.
- Common methods
  - SIFT (Scale-Invariant Feature Transform): Detects and describes local features in images.
  - HOG (Histogram of Oriented Gradients): Captures edge directions to describe appearance and shape.
- Example Application
  - Object recognition, where spatial features are crucial.

# Temporal features

- Capture motion information between consecutive frames.
- Common methods
  - Optical Flow: Measures the motion of objects between frames, used to understand movement.
  - Motion History Images (MHI): Represents motion by capturing the presence of motion in each pixel over time.
- Example Application
  - Recognizing dynamic actions like "running" or "jumping."

# Spatio-temporal features

- Combine spatial and temporal information to capture the evolution of actions over time.
- Common methods:
  - STIP (Space-Time Interest Points): Extends spatial interest points to the temporal domain.
  - 3D CNNs (3D Convolutional Neural Networks): Apply convolution across both spatial and temporal dimensions.
- Example Application:
  - Video classification where both spatial and temporal cues are important.

# Two-stream networks

- Consists of two separate streams: one for spatial (RGB images) and one for temporal (optical flow) information.
- The streams are usually combined (e.g., through late fusion) to produce the final action classification.
- Advantages:
  - Effectively captures both appearance and motion information.
- Example:
  - Two-Stream ConvNet (Simonyan and Zisserman, 2014):\*\* A groundbreaking model that processes spatial and temporal information separately.

# 3D Convolutional Networks

- Extends 2D convolution to the temporal dimension, processing entire video clips as input.
- Can capture both spatial and temporal features simultaneously.
- Advantages
  - Directly captures motion information without needing a separate optical flow calculation.
- Example
  - C3D (Tran et al., 2015) A deep 3D ConvNet that demonstrated the effectiveness of 3D convolutions for video analysis.

# Attention mechanisms

- Focus on the most relevant parts of the video frames, enhancing the model's ability to recognize important features.
- Types
  - Spatial Attention: Focuses on important regions within each frame.
  - Temporal Attention: Emphasizes key frames or moments in the sequence.
- Example:
  - Non-local Networks (Wang et al., 2018):\*\* Uses self-attention to capture long-range dependencies in video frames.

# Transformers for action recognition

- Transformers, originally designed for NLP, have been adapted for video by treating video frames as sequences.
- Capture global relationships within and across frames.
- Advantages:
  - Superior at capturing long-range dependencies and interactions between frames.
- Example:
  - TimeSformer (Bertasius et al., 2021): A transformer-based model that processes video as a sequence of patches.



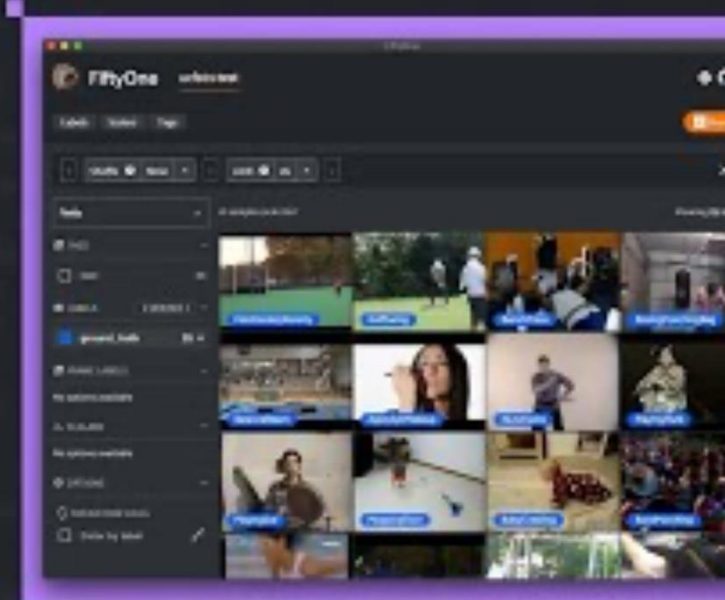
# UCF101

- A dataset with 13,320 video clips across 101 action categories.
- Categories include a wide range of actions like playing instruments, sports, and human-object interactions.
- Usage:
  - Standard benchmark for evaluating action recognition algorithms.
- Challenges:
  - Limited variability in terms of background and camera angles.
- <https://www.crcv.ucf.edu/research/data-sets/ucf101/>

# UCF101

## UCF101 Action Recognition Dataset

A human action recognition dataset collected from YouTube



# Kinetics

- A large-scale dataset containing approximately 650,000 video clips covering 400-700 human action classes.
- Clips are sourced from YouTube and cover a diverse set of human activities.
- Usage
  - Widely used for training and evaluating deep learning models.
- Challenges:
  - High variability and large scale, making it suitable for deep learning models.
- <https://github.com/cvdfoundation/kinetics-dataset>

# HMDB51

- A dataset with 6,766 video clips from 51 action categories, providing a challenging benchmark with a wide range of actions.
- Usage:
  - Often used alongside UCF101 for benchmarking action recognition models.
- Challenges:
  - Contains more complex actions with greater variability in performance.
- <https://serre-lab.clips.brown.edu/resource/hmdb-a-large-human-motion-database/>

# AVA (Atomic Visual Actions)

- Focuses on action recognition with spatio-temporal localization.
- Consists of over 430 video clips annotated with atomic actions, each annotated with bounding boxes and action labels.
- Usage:
  - Benchmark for tasks requiring precise localization of actions within a scene.
- Challenges:
  - Complex annotation and need for spatio-temporal understanding.
- <https://research.google.com/ava/>

# AVA (Atomic Visual Actions)

AVA [Dataset](#) [Explore](#) [Download](#) [Challenge](#) [About](#)

## Vertical

All

## Filter

## Entities

**stand (45790)** sit (30037)

talk to (e.g., self, a person, a group) (29020)

watch (a person) (25552)

listen to (a person) (21557)

carry/hold (an object) (18381) walk (12765)

bend/bow (at the waist) (2592) lie/sleep (1897)

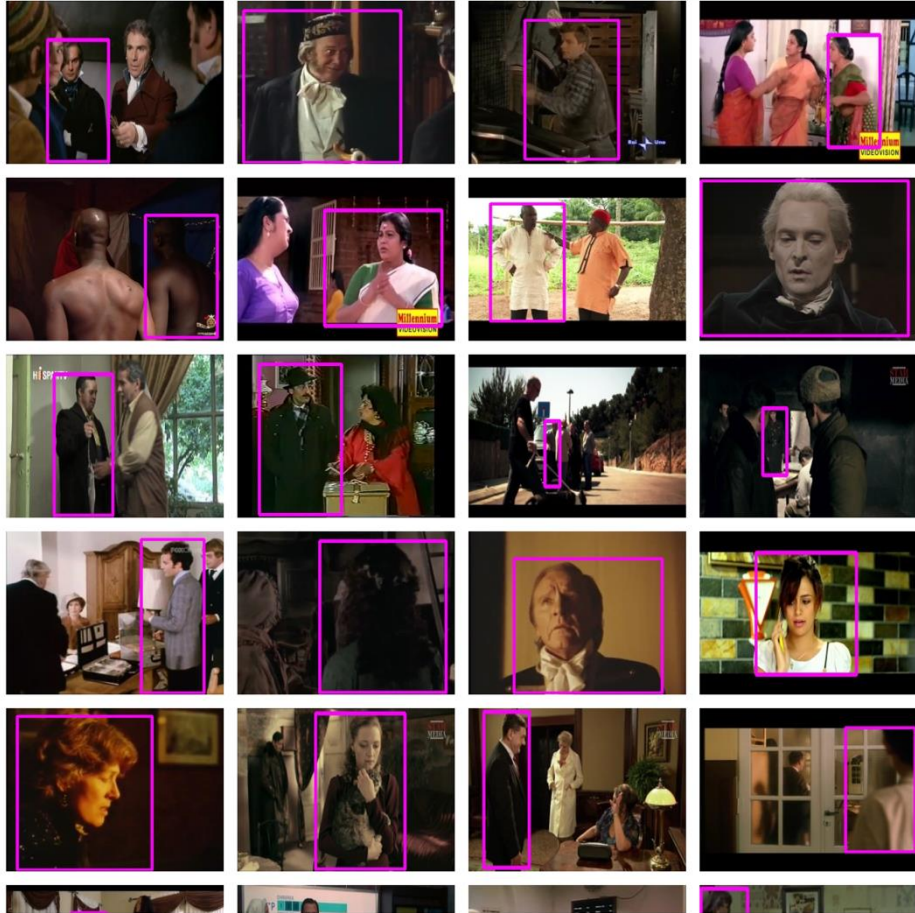
dance (1406)

ride (e.g., a bike, a car, a horse) (1344)

run/jog (1146) answer phone (1025)

watch (e.g., TV) (993) grab (a person) (936)

smoke (860) eat (828) fight/hit (a person) (707)



# Something-Something

- A dataset focused on fine-grained human-object interactions, challenging models to recognize subtle differences in actions.
- Usage:
  - Ideal for testing models on complex action recognition tasks



# Action Recognition Metrics

- Action Recognition tasks involve identifying and classifying human actions.
- Evaluating model performance is crucial.
- Common metrics are used to assess model effectiveness.



# Accuracy

- Definition: Proportion of correctly classified actions.
- Formula:

$$\text{Accuracy} = (\text{Number of Correct Predictions}) / (\text{Total Number of Predictions})$$

- Usage: Simple and intuitive, but may not reflect performance well with imbalanced classes.

# Precision

- Definition: Measure of the accuracy of positive predictions.
- Formula:

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP})$$

- Usage: Useful when the cost of false positives is high.

# Recall (Sensitivity)

- Definition: Measure of the ability to correctly identify positive instances.
- Formula:

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$$

- Usage: Important when the cost of false negatives is high.

# F1 Score

- Definition: Harmonic mean of precision and recall.

- Formula:

$$\text{F1 Score} = 2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$$

- Usage: Balances precision and recall, useful in imbalanced datasets.

# Confusion Matrix

- Definition: A table that describes the performance of a classification model.
- Structure:
  - True Positives (TP), False Positives (FP)
  - True Negatives (TN), False Negatives (FN)
- Visual: Include a 2x2 matrix with labels for TP, FP, TN, FN.

# Mean Average Precision (mAP)

- Definition: Mean of average precision across all classes.
- Formula:

$$\text{mAP} = (1/N) * \sum(\text{AP}_i)$$

where  $\text{AP}_i$  is the average precision for class  $i$ .

- Usage: Common in object detection and video action recognition.

# Top-k Accuracy

- Definition: Measures the percentage of samples where the correct label is among the top k predictions.
- Formula:

Top-k Accuracy = (Number of Correct Predictions in Top-k) / (Total Number of Predictions)

- Usage: Useful in multi-label or ambiguous classification tasks.

# Area Under the Curve (AUC - ROC)

- Definition: Measures the model's ability to distinguish between classes.
- Formula:

$$\text{AUC} = \int(\text{TPR}(\text{FPR}) \, d\text{FPR})$$

where TPR is True Positive Rate and FPR is False Positive Rate.

- Usage: Higher AUC indicates better model performance.



# Average Recall (AR)

- Definition: Average of recall scores across different IoU thresholds.
- Formula:

$$AR = (1/M) * \sum(\text{Recall}_i)$$

where M is the number of IoU thresholds.

- Usage: Common in temporal action detection tasks.

# Per-class Accuracy

- Definition: Accuracy computed for each class separately.
- Formula:

$$\text{Per-class Accuracy}_i = (TP_i + TN_i) / (TP_i + TN_i + FP_i + FN_i)$$

- Usage: Helps understand model performance on specific actions.

# Edit Distance

- Definition: Measures the similarity between predicted and ground truth action sequences.
- Formula:

$$\text{Edit Distance}(A, B) = \text{min\_operations}(A \rightarrow B)$$

- Usage: Important in tasks like temporal action segmentation.

# Intersection over Union (IoU)

- Definition: Measures the overlap between predicted and ground truth action segments.
- Formula:

$$\text{IoU} = (\text{Area of Overlap}) / (\text{Area of Union})$$

- Usage: Crucial in tasks requiring precise localization.

# Conclusion

- Encourage applying these metrics in action recognition tasks.

# How do I get an A+ in this course?

From ChatGPT

- Understand the Course Objectives
- Stay Consistent with Coursework
- Master the Theoretical Concepts
- Hands-on Practice
- Stay Updated
- Seek Feedback
- Form Study Groups
- Utilize Resources
- Manage Your Time
- Prepare for Exams
- Work on Projects with Passion
- Engage Beyond the Classroom

Lastly, always maintain a positive and curious mindset. Be proactive in your learning and seek opportunities to apply what you've learned. Remember, the ultimate goal is not just the A+ grade but gaining a deep understanding of computational photography and its applications.