PROJECT-PROPOSAL

Alessandro Tommasi – s353532
Alessio Sorrentino – s353528
Francesco Sicilia – s354909

Predictive Safety and Dynamic Risk Estimation in Human–Robot Collaboration

---

1. Dataset Description

The project is based on the public dataset available at:
https://github.com/AlessioSam/CHICO-PoseForecasting

The dataset consists of `.pkl` files containing motion sequences recorded during human–robot collaboration tasks. It includes:

- **20 subjects** (S00–S19) performing realistic industrial actions
- **7 action types**
- **24 joints per frame**, each defined by 3 coordinates (x, y, z) in millimeters:
    - 15 human skeleton joints
    - 9 robotic arm joints
- For each action, both **Normal** and **CRASH** versions are provided (with and without physical collision)

The dataset split follows exactly the protocol described in section 6.1 "Pose Forecasting Benchmark" of the reference document (https://arxiv.org/abs/2208.07308). In particular:

cit: “Evaluation protocol. We create the train/validation/test split by assigning 2 subjects to the validation (subjects 0 and 4), 4 to the test set (subjects 2, 3, 18 and 19), and the remaining 14 to the training set.”

The dataset is therefore composed as follows:

- Train: S01, S05, S06, S07, S08, S09, S10, S11, S12, S13, S14, S15, S16, S17
- Value: S00, S04
- Test: S02, S03, S18, S19

This subdivision complies with the criteria reported in the paper, ensuring consistency with the reference scientific methodology.

---

2. Risk Estimation Criterion

Each frame can be assigned a **risk class**, determined by computing the **Euclidean distance** between all possible human–robot joint pairs and considering the **worst case**. In practice, for each frame we take the **minimum distance** among all human–robot joint pairs; this value defines the **risk level**. Based on this distance, we use **three risk classes**, and each class corresponds to a specific label:

- **Class 0 – Safe:** distance > 630 mm
- **Class 1 – Near-collision:** 130 mm < distance ≤ 630 mm
- **Class 2 – Collision:** distance ≤ 130 mm

For each frame, the **minimum distance** between any human–robot joint pair defines the risk level

This criterion follows the approach used in the reference paper:
https://arxiv.org/abs/2208.07308

---

3. Architectures

Handling Class Imbalance and Safety-Critical Evaluation

By analyzing the dataset, we noticed that it is imbalanced, since most of the samples are labeled as safe. We use a **Weighted Cross Entropy Loss**. We assign a higher weight to the error when the target frame represents danger.

- **Safe:** weight 1.0
- **Near-Collision:** weight 5.0
- **Collision:** weight 20.0

This forces the network to prioritize dangerous trajectories, even if they are fewer than safe ones.

Metrics for Dangerous Classes:

- **Recall on the "collision" class:** the most critical safety metric (measures how many real collisions we detect).
- **Confusion Matrix:** shows how many collision cases are misclassified as safe.
- **Precision-Recall Curves:** computed specifically for the collision class to evaluate the trade-off between false alarms and missed dangers.

Phase 1 – MLP for risk classification

A **Multilayer Perceptron (MLP)** is used, ending with a softmax output layer.

- **Input**: coordinates of all 24 joints (x, y, z) in a single frame
- **Architecture:** fully connected layers with softmax output
- **Output**: predicted risk class (0, 1, or 2)
- **Loss:** cross-entropy (classification)

The dataset is preprocessed by computing risk labels using the human–robot distances, and then split into training, validation, and test sets.

Phase 2 – LSTM for future pose prediction

The second phase introduces a temporal prediction model based on **LSTMs**. We would like to use our network as a "**risk predictor**" composed as this:

- **LSTM Input:** a sequence of past frames
- **LSTM Output:** produces a vector of predicted risk-level class scores (one per class).
- **Loss:** cross-entropy (classification)

---

4. Training SetUp

Phase 1 – Classification

Since this is a multiclass classification task, the dataset is **enriched** with risk labels computed through distance measurements.

Phase 2 – Future pose prediction

The LSTM is trained to predict future frames using the **ground-truth joint positions already available in the dataset**.

---

5. Evaluation metrics

The networks will be evaluated using **confusion matrix**, **accuracy**, **precision**, **recall** and **F1-score.** Therefore, we'll analyze LSTM inputs that lead to the **highest MSE values**, with the aim of identifying: **recurrent error patterns**, **motion types that are more difficult to predict**, **factors that cause prediction error**.