Fasi della creazione del prototipo:

1)Creazione dataset:

1. Download del dataset RIS (usato per rimuovere la pioggia dalle immagini) da https://github.com/panda-lab/Single-Image-Deraining
2. Download dataset FloodIMG: Flood Image DataBase System (Kaggle) e selezione delle prime 3000 immagini
3. Download di 3000 immagini di Barcelona da GSV-Cities dataset (kaggle) (selezionato come buon tempo)
4. Labeling delle 3 categorie come good weather-rainy-flooding
5. Riduzione delle dimensioni in 224x224 e caricamento su kaggle (https://www.kaggle.com/datasets/marco0000000/flooding-rain-normal-weather-dataset)

2)Allenamento modello:

1. Labeling per la multiclassificazione usando one hot vector
2. Normalization of the image
3. Data augmentation(da migliorare)
4. Trasfer learning importando mobilenet-v2 con pesi di imagenet
5. Aggiunta di batch normalization ,2 dense layer ,1 dropout e 3 neuroni finali per l’output al fine di classificare le 2000 feature fornite da imagenet
6. Allenamento sul train , salvataggio del modello allenato , modello con minor loss e maggior accuracy e confronto finale - attualmente 91.33% di accuracy nel test

3)Inclusione del modello in un programma su raspberry

1. Creazione di un programma python che acquisisce immagini dalla webcam nativa tramile la libreria libcam (non compatibile con virtual environment e quindi utilizzabile solo da ambiente globale) e fornisce lo streaming video e le rivelazioni tramite un server http a singola pagina sulla porta 80
2. Uso del modello per la predizione su un frame ogni 2 minuti e cambio del colore di un semaforo a tre colori posizionato sull’immagine visibile in streaming se la predizione fornisce valori superiori all’80% per evitare oscillazioni tra i diversi grado di pericolo di allagamento rilevato
3. Tempo di inferenza del modello calcolato usando due chiamata a time.time() prima e dopo la predizione oscillante tra 300 e 430 millisecondi