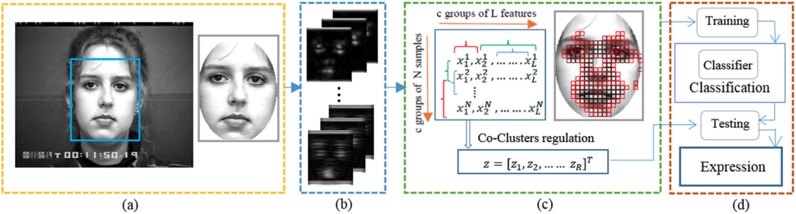
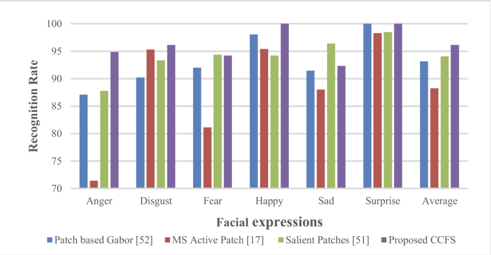
Co-Clustering to Reveal Salient Facial Features for Expression Recognition

- **Proposal** Use co-clustering to select features that can be used to classify facial expressions
- **Idea**
  - Use Gabor filter to extract features vectors from facial images
  - Use co-cluster to build a relationship chain
    Class $\overset{\text{label}}{\sim}$ Samples $\overset{\text{co-cluster}}{\sim}$ Features
- **Method**
  - Use Gabor filter to extract features from facial images
  - Use co-clustering to attain a subset of features and samples
  - Find the probability that a co-cluster is related to a certain class
  - Features with high probability are selected

Co-Clustering to Reveal Salient Facial Features for Expression Recognition





My Proposal

- **Proposal**: Use co-cluster to select features in NLP
- **Aims**: Use fewer features to get good prediction results, reducing the time and computational overhead of NLP training.
- **Method**:
  - Word-embedding or n-gram to extract features from text
  - (Maybe) Generate more features from the results above
  - Use co-clustering to attain a subset of features and samples
  - Select features that are related to a certain class most

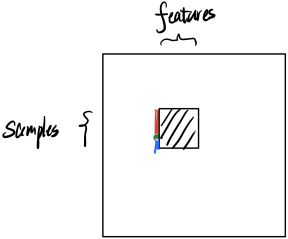


Figure: My Proposal, colors for classes

Understanding Black-box Predictions via Influence Functions (ICML 2017 Best Paper)

- **Proposal**: Influence function measures the impact of training data on the model - selection features
- **Method**:
  - Influence function of samples on model parameters: Formula 1
  - The influence function of samples on loss: Formula 2
  - The influence function of samples on the selection of features: formula 3
- **Discussion**: Different methods (co-cluster & influence function) for the same aims, need performance comparison

Method:

- Influence function of samples on model parameters: Formula 1
- The influence function of samples on loss: Formula 2
- The influence function of samples on the selection of features: formula 3

$$z_\delta \stackrel{\text{def}}{=} (x+\delta, y), \quad \hat{\theta}_{\epsilon,z} \stackrel{\text{def}}{=} \arg\min_{\theta\in\Theta} \frac{1}{n}\sum_{i=1}^{n} L(z_i,\theta) + \epsilon L(z,\theta)$$

$$\hat{\theta}_{\epsilon,z_\delta,-z} \stackrel{\text{def}}{=} \arg\min_{\theta\in\Theta} \frac{1}{n}\sum_{i=1}^{n} L(z_i,\theta) + \epsilon L(z_\delta,\theta) - \epsilon L(z,\theta)$$

$$\mathcal{I}_{\text{up,params}}(z) \stackrel{\text{def}}{=} \frac{d\hat{\theta}_{\epsilon,z}}{d\epsilon}\Big|_{\epsilon=0} = -H_{\hat{\theta}}^{-1}\nabla_\theta L(z,\hat{\theta})$$

$$\begin{aligned}\mathcal{I}_{\text{pert,loss}}(z, z_{\text{test}})^\top &\stackrel{\text{def}}{=} \nabla_\delta L(z_{\text{test}}, \hat{\theta}_{z_\delta,-z})^\top\Big|_{\delta=0} \\ &= -\nabla_\theta L(z_{\text{test}},\hat{\theta})^\top H_{\hat{\theta}}^{-1}\nabla_x\nabla_\theta L(z,\hat{\theta})\end{aligned}$$