Reinforcement Learning for Active Appearance Model Dataset Selection

André Antonitsch
School of Technology
PUCRS

Introduction

Active Appearance Model (AAM) is an example of an “Interpretation through synthesis” approach [Cootes, Edwards, and Taylor 1998; 2001]

The main goal is to describe novel images by generating a synthetic image which is as similar as possible, using a parameterized model.

Among possible motivations:

- robust image segmentation;
- appearance codification into compact parameters;
- reducing the number of parameters needed for appearance description;
- and so on

Introduction

The quality of an AAM is directly correlated with how expressive the dataset used to model it is.

- That is, the vector space it describes is as large as possible.

However, the time it takes to train an AAM is also directly correlated with how large a dataset is.

Which is an issue when dealing with video originated datasets, for exemple.

- These contain lots of sequential images, which tend to be redundant.

Introduction

Reinforcement Learning is an approach for learning policies, that is a set of actions to take based on a certain current state of the environment.

Reinforcement Learning seeks to find the optimal policy under conditions which are stochastic or hard to predict. One of such algorithms is Q-Learning.

The dataset selection task is similar to the task of optimizing databases.

One of such reinforcement learning approaches for databases is proposed by Basu et al. (2016)

Introduction

In this project we propose a Reinforcement Learning approach to select a sub-set of a larger dataset for use with AAM.

We seek to provide a tradeoff of training speed with minimal loss of expressiveness as possible.

Technical Approach - Modeling Appearance

The AAM statistical model is generated by combining a model of shape and a model of texture.

To build the model, a training set of annotated images with corresponding landmarks is required.

The model builds a mean-shape vector and for each image in the dataset, computes its shape-variation in relation to the mean. It then builds a co-variance matrix of the shape vectors.

Technical Approach - Modeling Appearance

By applying PCA to the obtained co-variance matrix, it is possible to parameterize each possible appearance with parameters **c**, controlling shape and texture of an appearance.

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{Q}_s \mathbf{c}$$

Where $\bar{\mathbf{x}}$ is the mean shape and $\mathbf{Q}_s$ is the the matrix describing the modes of variation in the training set (and the eigenvectors of the model).

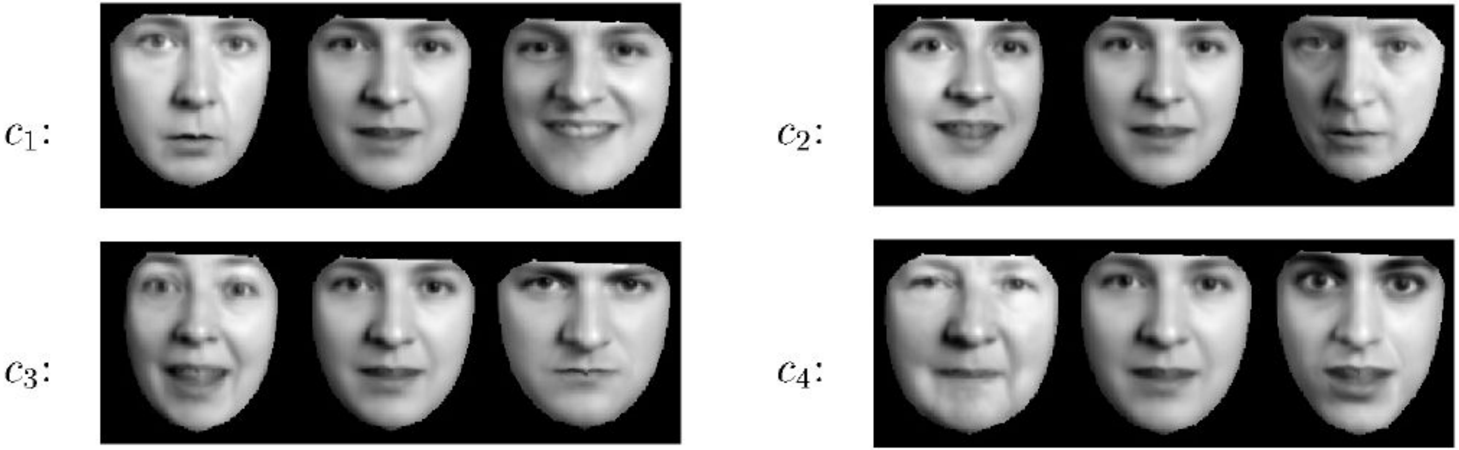
Technical Approach - Modeling Appearance

This allows the model to generate images not in the original dataset.

But it still assumes the original dataset vectors to define an Allowed Shape Space.

- That is, the space where shapes are assumed to be correct.

Technical Approach - Modeling Appearance



The effects of varying the first four appearance model parameters, by ± 3 standard deviations from the mean.

Extracted from [Cootes, Edwards, and Taylor 1998; 2001]

Technical Approach - Reinforcement Learning

Reinforcement learning is an approach to how agents can learn optimal policies for action taking in a given environment (Sutton and Barto 2018; Kaelbling, Littman, and Moore 1996).

Generally, reinforcement learning problems are modeled as a Markov Decision Process.

And the objective is to learn the optimal policy which leads agentes to maximize their accumulated reward given a certain reward function.

For this project we plan to use Q-Learning.

Technical Approach - Q-Learning

Q-Learning is an iterative algorithm for policy learning.

Q-Learning fills a look-up table with value-pairs: state-action and reward associated.

- For each given state s and a given action a, the algorithm associates a reward value.

This algorithm has the problem that filling this look up table can be time consuming.

Technical Approach - Q-Learning

Tsitsiklis and Van Roy (1997) propose using Q-Learning to learn a linear interpolation function, instead of a look-up table.

And with each update step of the Q-Learning algorithm, they propose updating the weights this linear interpolation function gives to each feature in the dataset, instead of associating it to a value-pair.

Technical Approach - Method and Evaluation

For this project we propose using a linearly interpolated Q-Learning algorithm for database selection for AAMs.

To achieve this goal we can consider our state space the set containing all the sub-sets of images in a database containing a specific number of images.

Technical Approach - Method and Evaluation

To train the Q-Learning algorithm, we consider each state a set of images $\{i_n, i_j ... i_z\}$

And represent the state vector for state a vector f = $\{i_{nx}, i_{jx} ... i_{zx}, i_{ny}, i_{jy} ... i_{zy}\}$

And associate a vector w of weights to the linear interpolation function.

By learning the weights of w we can compute the value of a given state by :

$$v = f \cdot w^T$$

Technical Approach - Method and Evaluation

The evaluation of the models can be done in a few different manners:

- Ideally, by testing the AAM generated using a image fitting procedure, seeking the least error measure in a testing dataset.
- In case that takes too long, we can evaluate how many eigenvectors and how significant the modes of variation are generated with a certain sub-dataset.

Conclusions

- In this project we propose a dataset selection algorithm using Q-Learning for AAM training.
- We plan to investigate whether the linear interpolation is the right fitting function for this task, as well as how impactful the reduced dataset is in terms of reduced time for model training and how expressive the reduced model is in relation to the original.
- Additionally, we expect to see large reductions in training time for video based datasets.

References

Cootes, T. F.; Taylor, C. J.; Cooper, D. H.; and Graham, J. 1995. Active shape models-their training and application. *Computer vision and image understanding* 61(1):38–59.

Cootes, T. F.; Edwards, G. J.; and Taylor, C. J. 1998. Active appearance models. In *European conference on computer vision*, 484–498. Springer.

Cootes, T. F.; Edwards, G. J.; and Taylor, C. J. 2001. Active appearance models. *IEEE Transactions on Pattern Analysis & Machine Intelligence* (6):681–685.

Kaelbling, L. P.; Littman, M. L.; and Moore, A. W. 1996. Reinforcement learning: A survey. *Journal of artificial intelligence research* 4:237–285.

Sutton, R. S., and Barto, A. G. 2018. *Reinforcement learning: An introduction*.

Tsitsiklis, J. N., and Van Roy, B. 1997. Analysis of temporal-diffference learning with function approximation. In *Advances in neural information processing systems*, 1075–1081.

Basu, D.; Lin, Q.; Chen, W.; Vo, H. T.; Yuan, Z.; Senellart, P.; and Bressan, S. 2016. Regularized cost-model oblivious database tuning with reinforcement learning. In *Transactions on Large-Scale Data-and Knowledge-Centered Systems XXVIII*. Springer. 96–132.