Learning Equivariant Structured Output SVM Regressors
by Vedaldi, Blaschko, Zisserman. ICCV 2011.
Interpreted by Magnus Burenius, KTH.
Invariance to Transformations

- **Pose-invariant classification.** Recognize an object category *regardless* of the object translation, rotation, and scale.

- **Pose regression.** Detect an object and estimate its translation, rotation, and scale.

- **Detection.** Find an object location (center), *without* estimating its orientation and scale.
Invariance and Equivariance

- Consider some transformation, like rotation.
- We would like object classification to be invariant to rotation.
- We would like object detection to be equivariant to rotation.
The Problem

- Input space: \( X \)
- Output space: \( Y \)
- Consider a transformation \( t \) acting on the input and output:
  \[ t = (t_X, t_Y) \in T \]
  \[ t_X : X \rightarrow X \]
  \[ t_Y : Y \rightarrow Y \]
- We want to learn a predictor \( f(x, w) \) that is invariant or equivariant to the transformation \( t \):
  \[ f(t_X x; w) = f(x; w) \quad \text{invariance} \quad (t_Y = I) \]
  \[ f(t_X x; w) = t_Y f(x; w) \quad \text{equivariance} \]
The Problem

• Most of the time we do not have enough training data, representing all possible transformations.
One Approach

- Can generate more training data by transforming the original data.
- How many samples should be generated? How densely? What samples are relevant?
Another Approach

- Could explicitly model and estimate transformations as latent variables.
- Learning problem becomes non-convex. Inference might be slower.
Their Approach

- Generalize Structured SVM to incorporate invariance and equivariance into a convex training procedure.
- Removes the need for ad-hoc sampling strategies. Only generates the virtual samples that are necessary.
- Inference does not require the estimate of latent variables.
Toy Example

Assuming rotation invariance

\[ X = R^2 \]
\[ Y = \{ r, g, b \} \]

Gradually enforce invariance to larger rotations →
Standard Structured SVM

- Let $X$ and $Y$ be the input and output and let $\Omega_X$ and $\Omega_Y$ be their sample spaces. These can be ANY spaces, not just integers or real vector spaces.

- A feature function $\Psi$ is used to map a pair from these complicated spaces to something we can compute with:

$$\Psi : \Omega_X \times \Omega_Y \to \mathbb{R}^d$$
Standard Structured SVM

• A classifier described by a vector $\omega$ predicts a class by solving

$$f(x; \omega) = \arg\max_y \omega \cdot \Psi(x, y)$$

• This imposes a restriction on $\Psi$
Standard Structured SVM

During training the STRUCTURE of the output space is taking into account by defining a loss function

$$\Delta : \Omega_y \times \Omega_y \rightarrow R$$

which quantifies the loss of predicting $y_p$ when the true output is $y$. It should fulfill

$$\Delta (y, y_p) \geq 0$$

$$\Delta (y, y_p) = 0 \text{ iff } y = y_p$$

$\Delta$ should thus reflect the quantity which measures how well the classifier performs.
Standard Structured SVM

- Given a training set \((x_1, y_1) \ldots (x_N, y_N)\) of "only positives" and a regularization constant \(C\) a classifier \(\omega\) is trained by solving the convex optimization problem:

\[
\min_{\omega} \|\omega\|^2 + C \sum_{n} \max_{y} \left( \Delta(y_n, y) + \omega \cdot \Psi(x_n, y) - \omega \cdot \Psi(x_n, y_n) \right)
\]

Search for difficult classifications
Their Generalization of S-SVM

Standard S-SVM:
\[
\min_{\omega} \| \omega \|^2 + C \sum_{n} \max_{y} \left( \Delta(y_n, y) + \omega \cdot \Psi(x_n, y) - \omega \cdot \Psi(x_n, y_n) \right)
\]

Transformation equivariant generalization:
\[
\min_{\omega} \| \omega \|^2 + C \sum_{n} \max_{y, t} \left( \Delta(t, y_n, y) + \omega \cdot \Psi(t_X x_n, y) - \omega \cdot \Psi(t_X x_n, t_Y y_n) \right)
\]

When looking for the difficult classifications we search over all possible equivariant variations of input and output.
Training

• The problem can be optimized using standard S-SVM solvers.
• These solvers handle the large number of constraints by generating the necessary ones on the fly.
• This corresponds to generating relevant virtual training data.
Advantages

- Principled approach to the generation of relevant virtual training data.
- Training is convex and no more expensive than standard Structured SVM and latent SVM.
- Inference is faster than latent SVM, since the latent variable, corresponding to the transformation, is not estimated.
Experiment 1
Rotation Equivariant Object Detection

- Let $\Phi(x,y)$ be the HOG-descriptor of a block of 7x7 HOG-cells at position y in the image x.
- A linear HOG model is not sufficient to capture arbitrary object rotations.
- They use something they call “slot kernel”.
- The cluster the HOG-space into $Q=18$ clusters.
- The total feature function is the outer product:

$$\Psi(x, y) = \Phi(x, y) e_q^T(\Phi(x, y))$$
Experiment 1 - Results

Aerial car detection. 30 images having a total of 1000 cars with different rotations. Unclear division of training and test data.
Motion as Natural Transformations

- Consider pedestrian detection in video.
- Training data consists of many sequences of moving persons.
- The frames from the same sequence are highly correlated.
- This breaks the assumption of i.i.d. samples, which is fundamental for most machine learning methods.
Motion as Natural Transformations

- Consider a sequence as a single training sample, and the different frames in it as transformations of it.
Experiment 2
Pedestrian Classification

- DaimlerChrysler pedestrian classification benchmark. The training data consists of 800 positive images and 5000 negative images, and two test sets of the same size.
- Consider mirroring and translation by 1 pixel as transformations.
- Also consider motion as natural transformations.
- They derive and compare an invariant binary SVM and an invariant rank rank SVM.
Experiment 2

(a) A graph showing the true positive rate against the false positive rate for different jitter conditions: no jitter, flip left-right, all jitters.

(b) A bar chart showing the 100 - EER (Error Rate) for different sizes of groups.

(c) Another bar chart showing the number of samples selected for different sizes of groups.
Conclusion

• The authors propose the use of their algorithm instead of ad-hoc sampling strategies or latent variables to incorporate invariance and equivariance.