Object Detection Grammars

Pedro Felzenszwalb
Brown University

Joint work with Ross Girshick and David McAllester
The challenge

Objects in each category vary greatly in appearance
An evolution of models

- HOG features/templates [Dalal, Triggs 2005]
  - Invariance to photometric variation and small deformations + SVM training

- Deformable part models (DPM)
  - HOG templates + LSVM training
  - Invariance to larger deformations

- Mixtures of DPM
  - Allow significant variations due to major poses and subtypes
Deformable models

• Can take us a long way...

• But not all the way
Structure variation

- Object in rich categories have variable structure

- These are NOT deformations

- Mixture of deformable models? too many combined choices
  - There is always something you never saw before

- Bag of words? not enough structure

- Non-parametric? doesn’t generalize
Richer part-based models

• Some parts should be optional
  - A person could have a hat or not

• There should be subtypes (mixtures) at the part level
  - A person could wear a skirt or pants
  - A mouth can be smiling or frowning

• Parts should be reusable
  - A wheel model can be used twice in a car model
  - Same wheel model can be used in car and truck model et

• This can be done using a grammar/compositional model
  - [Jin, Geman, 2006], [Zhu, Mumford, 2006], [Zhu, Yuille, 2005], etc.
Object detection grammars

(A tractable compositional framework)

• Objects defined in terms of other objects through production rules
  - face -> eyes, nose, mouth

• Objects can be defined by multiple productions
  - legs -> pants
  - legs -> skirt
  - Subtypes, structure variability

• Deformation rules allow parts to move relative to each other
  - Spatial variability

• Same object can be used in different productions
  - Shared parts
- person -> face, trunk, arms, lower-part
- face -> eyes, nose, mouth
- face -> hat, eyes, nose, mouth
- hat -> baseball-cap
- hat -> sombrero
- lower-part -> shoe, shoe, legs
- lower-part -> bare-foot, bare-foot, legs
- legs -> pants
- legs -> skirt
Relationship to pictorial structures / DPM

- Pictorial structure
  - parts (local appearance)
  - springs (spatial relationships)
  - parts and springs forms a graph --- structure is fixed

- Object detection grammar
  - Grammar generates tree of symbols --- structure is variable
  - Location of symbol is related to location of parent
  - Appearance model associated with each terminal
Formalism

- Set of terminal symbols $T$
  - (templates)
- Set of nonterminal symbols $N$
  - (objects/parts)
- Set of placements $\Omega$ within an image
- Placed symbol $X(\omega)$
  - $X \in T \cup N$
  - $\omega \in \Omega$

$\omega$ might be (x,y) position and scale

face((90,10),50)

eye((100,80),10)
Production rules

• Productions define expansions of nonterminals into bags of symbols

\[ X(\omega) \rightarrow s \rightarrow \{ Y_1(\omega_1), \ldots, Y_n(\omega_n) \} \]

placed nonterminal score Bag of placed symbols

• We can expand a nonterminal into a bag of terminals by repeatedly applying productions
  - There are choices along the way
  - Expansion has score = sum of scores of productions used along the way
  - \( X(\omega) \rightarrow s \rightarrow \{ A_1(\omega_1), \ldots, A_n(\omega_n) \} \) (sequence of expansions)
  - Leads to a derivation tree
Appearance for terminals

- Each terminal has an appearance model
  
  Defined by a scoring function $f(A, \omega, I)$
  
  Score for placing terminal $A$ at position $\omega$ within image $I$

$f(A, \omega, I)$ might be the response of a HOG filter $F_A$ at position $\omega$ within $I$
Appearance for nonterminals

• We extend the appearance model from terminals to nonterminals

\[ f(X, \omega, l) = \max_{X(\omega) \sim s \sim \rightarrow \{ A_1(\omega_1), \ldots, A_n(\omega_n) \}} (s + \sum_i f(A_i, \omega_i, l)) \]

• Best expansion of \( X(\omega) \) into a bag of placed terminals

  - Takes into account

    1) expansion score

    2) appearance model of placed terminals at their placements

• Detect objects (any symbol) by finding high scoring placements
Implementation

• General implementation for a class of grammars (voc-release4)
  - Production rules specified by schemas
  - Appearance of terminals defined by HOG filters
  - Inference done via dynamic programming
  - Parameter learning from bounding boxes (LSVM)
Isolated deformation grammars

• Productions defined by two kinds of schemas

• Structure schema
  - One production for each placement $\omega$

    \[ X(\omega) \rightarrow s \rightarrow \{ Y_1(\omega+\delta_1), \ldots, Y_n(\omega+\delta_n) \} \]

• Deformation schema
  - One production for each $\omega$ and displacement $\delta$

    \[ X(\omega) \rightarrow s(\delta) \rightarrow \{ Y(\omega+\delta) \} \]

• Leads to efficient algorithm for computing scores $f(X,\omega,I)$
Face grammar

\[ N = \{ \text{FACE}, \text{EYE}, \text{EYE}', \text{MOUTH}, \text{MOUTH}' \} , \]
\[ T = \{ \text{FACE}.\text{FILTER}, \text{EYE}.\text{FILTER}, \text{SMILE}.\text{FILTER}, \text{FROWN}.\text{FILTER} \} . \]
Face grammar

\[ N = \{ \text{FACE}, \text{EYE}, \text{EYE}', \text{MOUTH}, \text{MOUTH}' \}, \]
\[ T = \{ \text{FACE}.\text{FILTER}, \text{EYE}.\text{FILTER}, \text{SMILE}.\text{FILTER}, \text{FROWN}.\text{FILTER} \}. \]

1) Face defined by global template and parts
\[ \forall \omega : \text{FACE}(\omega) \xrightarrow{0} \{ \text{FACE}.\text{FILTER}(\omega), \text{EYE}'(\omega \oplus \delta_l), \text{EYE}'(\omega \oplus \delta_r), \text{MOUTH}'(\omega \oplus \delta_m) \}. \]
Face grammar

\[ N = \{ \text{FACE, EYE, EYE}', \text{MOUTH, MOUTH}' \}, \]
\[ T = \{ \text{FACE.FILTER, EYE.FILTER, SMILE.FILTER, FROWN.FILTER} \}. \]

1) Face defined by global template and parts
\[ \forall \omega : \text{FACE}(\omega) \xrightarrow{0} \{ \text{FACE.FILTER}(\omega), \text{EYE}'(\omega \oplus \delta_l), \text{EYE}'(\omega \oplus \delta_r), \text{MOUTH}'(\omega \oplus \delta_m) \}. \]

2) Parts can move relative to their idea location
\[ \forall \omega, \delta : \text{EYE}'(\omega) \xrightarrow{||\delta||^2} \{ \text{EYE}(\omega \oplus \delta) \}, \]
\[ \forall \omega, \delta : \text{MOUTH}'(\omega) \xrightarrow{||\delta||^2} \{ \text{MOUTH}(\omega \oplus \delta) \}. \]
Face grammar

\[ N = \{ \text{FACE, EYE, EYE}', \text{MOUTH, MOUTH}' \}, \]
\[ T = \{ \text{FACE.FILTER, EYE.FILTER, SMILE.FILTER, FROWN.FILTER} \}. \]

1) Face defined by global template and parts
\[ \forall \omega : \text{FACE}(\omega) \xrightarrow{0} \{ \text{FACE.FILTER}(\omega), \text{EYE}'(\omega \oplus \delta_l), \text{EYE}'(\omega \oplus \delta_r), \text{MOUTH}'(\omega \oplus \delta_m) \}. \]

2) Parts can move relative to their idea location
\[ \forall \omega, \delta : \text{EYE}'(\omega) \xrightarrow{\|\delta\|^2} \{ \text{EYE}(\omega \oplus \delta) \}, \]
\[ \forall \omega, \delta : \text{MOUTH}'(\omega) \xrightarrow{\|\delta\|^2} \{ \text{MOUTH}(\omega \oplus \delta) \}. \]

3) Parts defined by templates
\[ \forall \omega : \text{EYE}(\omega) \xrightarrow{0} \{ \text{EYE.FILTER}(\omega) \}, \]
\[ \forall \omega : \text{MOUTH}(\omega) \xrightarrow{s} \{ \text{SMILE.FILTER}(\omega) \}, \]
\[ \forall \omega : \text{MOUTH}(\omega) \xrightarrow{f} \{ \text{FROWN.FILTER}(\omega) \}. \]
Learning

\[ f(X, \omega, l) = \max_{X(\omega) \sim \sim s \sim \sim} \{ A_1(\omega_1), \ldots, A_n(\omega_n) \} \]

\[ f(X, \omega, l) = \max_{z} w^T \phi(z) \]

- \( z \) is an expansion of \( X(\omega) \) into a bag of terminals
- \( w \) is a vector of model parameters
  - Score of each structure schema
  - Deformation parameters of each deformation schema
  - Appearance template for each terminal (HOG filters)
- \( w \) can be trained using Latent SVM
Person detection grammar [NIPS 2011]

- Instantiation includes a variable number of parts
  - 1,...,k and occluder if k < 6
- Parts can translate relative to each other
- Parts have subtypes
- Parts have deformable sub-parts (not shown)
- Beats all other methods on PASCAL 2010 (49.5 AP)
Building the model

- Type in manually defined grammar

\[
Q(\omega) \xrightarrow{s_k} \{ Y_1(\omega \oplus \delta_1), \ldots, Y_k(\omega \oplus \delta_k), O(\omega \oplus \delta_{k+1}) \} \\
Q(\omega) \xrightarrow{s_6} \{ Y_1(\omega \oplus \delta_1), \ldots, Y_6(\omega \oplus \delta_6) \}
\]

\[
Y_p(\omega) \xrightarrow{0} \{ Y_{p,t}(\omega) \} \\
O(\omega) \xrightarrow{0} \{ O_t(\omega) \} \\
O_t(\omega) \xrightarrow{\alpha_t \cdot \phi(\delta)} \{ A_t(\omega \oplus \delta) \}
\]

\[
Y_{p,t}(\omega) \xrightarrow{\alpha_{p,t} \cdot \phi(\delta)} \{ Z_{p,t}(\omega \oplus \delta) \} \\
Z_{p,t}(\omega) \xrightarrow{0} \{ A_{p,t}(\omega), W_{p,t,r,1}(\omega \oplus \delta_{p,t,r,1}), \ldots, W_{p,t,r,N_p}(\omega \oplus \delta_{p,t,r,N_p}) \} \\
W_{p,t,r,u}(\omega) \xrightarrow{\alpha_{p,t,r,u} \cdot \phi(\delta)} \{ A_{p,t,r,u}(\omega \oplus \delta) \}
\]

- Train parameters from bounding box annotations
  - Production scores
  - Deformation models
  - HOG filters for terminals
Detections with person grammar

(a) Full visibility
(b) Occlusion boundaries

Figure: Example detections. Parts are blue. The occlusion part, if used, is dashed cyan.
(a) Detections of fully visible people. (b) Examples where the occlusion part detects an occlusion boundary.

Qualitative results 1

(a) Early termination
(b) Mistakes

Figure: Example detections. Parts are blue. The occlusion part, if used, is dashed cyan.
(a) Detections where there is no occlusion, but a partial person is appropriate. (b) Mistakes, where the model did not detect occlusion properly.
Evolution

HOG (DT, CVPR05)
AP=0.16

DPM (CVPR08)
AP=0.27

2 DPM (PAMI10)
AP=0.36

6 DPM (voc-release4)
AP=0.43

Grammar (NIPS11)
AP=0.47
Summary

• The big challenge is handling appearance variation

• Object detection grammars can express many types of models
  - Mixtures of DPM
  - Models with variable structure
  - Models with shared parts
  - etc. -- think of it as a programming language

• General implementation
  - Isolated deformation grammars + HOG + LSVM

• Learning grammar structure is still an open problem