Deep Learning for Visual and Virtual Worlds

Eleonora Vig, Ph.D.

Remote Sensing Technology Institute
German Aerospace Center (DLR)
Outline

1. Convolutional Neural Networks for image/video analysis
2. Two solutions to the problem of annotated data scarcity
3. Applications in aerial image and video understanding
Predictions on images and videos

Object detection

Segmentation

Action recognition

Video prediction

Object tracking
Predictions on images and videos – until 2012

The two-step pipeline:

input signal → feature extraction → representation → prediction → output prediction

Hand-designed features:
😊 domain specific, expert knowledge
😊 design is tedious, ad-hoc process
😊 restricted and subjective choice of features
Predictions on images and videos – after 2012

Convolutional Neural Network (CNN) – extracts more complex/abstract features *and* learns to perform the task at hand

from http://vision03.csail.mit.edu/cnn_art/index.html
Convolutional Neural Network (CNN) model zoo

- **AlexNet**: 138M parameters
- **VGG16**: 62M parameters
- **ResNet**: 25M parameters
- **DenseNet**: 40M parameters
- **GoogLeNet**: 23M parameters
2. Challenges

Setting millions of parameters ("training") → huge labeled datasets required

- **Problem: Lack of Labeled (Video) Data**
  - 😞 high data acquisition and labeling costs
  - 😞 limited quantity and variety: weather, rare events, etc.
  - 😞 project specificity

- **Solution 1: Synthetic Data from Modern Game Engines**
  - 😊 full control over data generation (quantity & variety)
  - 😊 virtual world creation is time-consuming

- **Solution 2: Domain Adaptation**
  - 😊 reusing and adapting “generic” models to be task-specific
Solution 1: Use of synthetic data

- Virtual KITTI dataset: **50 synthetic videos** generated from **5 virtual worlds** created in Unity game engine in urban settings under different imaging and weather conditions
- automatically and fully annotated for several video understanding tasks

Aside: KITTI Vision Benchmark Suite

Benchmark suite to test algorithms on real-world data found in autonomous driving (ADAS).

- **collection protocol**: two high-resolution cameras mounted on a moving vehicle + laser scanner + GPS/IMU sensors
- **tasks of interest**: visual odometry (localization), object detection and tracking, road/lane estimation, semantic segmentation, etc.
Generating proxy virtual worlds – Cloning

**Small “seed” real-world data:** image sequences, 3D Lidar point clouds, GPS/IMU, bounding boxes around objects

**Generating synthetic clones:**

▶ automatic placement of roads and vehicles from sensor data
▶ manual configuration of background: trees, buildings, sun, etc.
Application: Multi-Object Tracking (MOT)

Multi-object tracking:

- task: estimate object trajectories for target class instances (cars)
- challenging: crowded urban scenes + moving camera
- tracking-by-detection: build tracks by linking detections through time
Quantitative results

1. Training on virtual data improves real-world performance
   Training on:
   - 'r': real videos
   - 'v': corresponding virtual “clones”
   - 'v→r': clones, then real videos
   Evaluation on held-out real videos.

<table>
<thead>
<tr>
<th>Method</th>
<th>MOTA↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>DP-MCF r</td>
<td>71.9%</td>
</tr>
<tr>
<td>DP-MCF v</td>
<td>64.3%</td>
</tr>
<tr>
<td>DP-MCF v→r</td>
<td>76.7%</td>
</tr>
</tbody>
</table>

2. Real-world models transfer across the real-to-virtual gap
   - evaluation on real-world ('r') vs. virtual “clone” ('v') videos
   - minimal real-to-virtual gap: MOTA difference < 0.5%

<table>
<thead>
<tr>
<th>Method</th>
<th>MOTA↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>DP-MCF r</td>
<td>81.7%</td>
</tr>
<tr>
<td>DP-MCF v</td>
<td>82.2%</td>
</tr>
</tbody>
</table>

MOTA: Multi-Object Tracking Accuracy
DP-MCF: Dynamic Programming Min Cost Flow tracking
Qualitative results

Comparing real-world pre-trained models on real videos vs. synthetic clones
Other recent synthetic datasets

- Virtual KITTI: one of the first synthetic video datasets
- GTA: “Playing for data: Ground truth from computer games”, Richter et al., ECCV, 2016
- VIPER: “Playing for benchmarks”, Richter et al., ICCV, 2017
Solution 2: Domain Adaptation

Domain mismatch: PASCAL VOC (source) \(\leftrightarrow\) KITTI (target)

Solution: adaptation of existing “general-purpose” object detectors to specific applications
Domain Adaptation for Multi-Object Tracking

Our solution – perform adaptation:

▶ from category to instances: \( w \rightarrow w_i \)
▶ from instances back to the category: \( w_i \rightarrow w \)

Previous work has not investigated both jointly.
Reducing the source/target domain gap

- domain gap due to distribution change between source/target data → degrades performance
- domain adaptation helps to narrow this gap
- industrial applications: parking occupancy estimation from vehicle-mounted cameras, etc.
3. Applications in Aerial Image and Video Understanding
Aerial crowd monitoring at large-scale events

Goals:
- ward off catastrophes and monitor emergencies
- help planning of large-scale events
HD-Maps for autonomous driving

- enhance existing maps with meta-information about lane type, lane markings, parking lots, etc.
- validation of vehicle-based multi-sensor navigation by airborne imagery
- urban zoning classification, etc.
Object detection in unconstrained aerial & satellite images

Key component for traffic monitoring, parking-lot utilization, urban planning, disaster management, etc.

Future challenges and directions

Trends in Computer Vision:

- move away from strong-supervision (i.e. large labeled datasets) to weak-supervision (i.e. noisy/incomplete labels): incremental learning, few-shot learning
- **Auto-ML**: automatically designing novel architectures
- explainability/interpretability of network decisions
- reliability values for network decisions
- robustness against adversarial samples
- high-level reasoning is largely unsolved