Object detection & classification for ADAS

✓ Robust for Bad situations
✓ Small object sizes
✓ Robust for occlusion
✓ Small model size
SVNet @ NVIDIA TX2

Please click Icon for Video
Robust detection for various situations

Snow

Night w/ Lamp

Rain

Fog

Please click Icon for Video

Please click Icon for Video

Please click Icon for Video

Please click Icon for Video
SVNet Algorithm Flow

Conv Layer
- Conv layer: deep convolutional neural networks

Proposal Layer
- Proposal layer: multi-scale region proposal

FC Layer
- FC layer: Fully Connected networks

Feature map
- Feature vectors

Candidate Regions
- Robust for Bad situations
- Small object sizes
- Robust for occlusion
- Small model size

Detection Results
- (Bonding Box, label)

✓ optimal parameters of network (size of kernels, # of layers, depth of channels) for the target platform
✓ optimal parameters of network (# of layers, weight connections) for the target platform
Manual Correction on 5% of the objects in input images

- Pedestrian: 94%, Vehicle: 95%
- Pedestrian: 6%, Vehicle: 5%
- ~1 in 5 min video

Input image → Automatic Labeling → Detection Success → Ground Truth

Automatic Labeling → Detection Failure → Manual Correction

Detection Failure → False Detection

Pedestrian: 94%, Vehicle: 95%
Pedestrian: 6%, Vehicle: 5%
~1 in 5 min video

Manual Correction on 5% of the objects in input images
How we use GPU (Titan X and GTX1080) for training

- **Models**
  - Designed by human experts

- **Target H/W**
  - Where we measure speed to select candidates before training

- **GPUs**
  - Train candidate models & evaluate their accuracy

- **Road Test**
  - Pass <10%
  - Pass ~30%
  - Start from >50 prototypes

- **GPU utilization last month**

- **~3 hours**
- **~2 days**
- **~2 weeks**
- **~2 months**
CuDNN framework

FP16 - HGEMM

cuDNN  cuBLAS

FP16 Intrinsics

hadd hsub hneg hmul hrcp hexp hgtu heq …

Lower memory bandwidth
Faster kernel execution

(*) Image from https://devblogs.nvidia.com/parallelforall/jetson-tx2-delivers-twice-intelligence-edge/
Customized Development Examples

Example: Collision Warning at Blind Corner using PD/VD on Curved Mirror
Publications

- **Local Decorrelation for Improved Pedestrian Detection**
  *Woonhyun Nam, Piotr Dollár, and Joon Hee Han.*  

- **Macrofeature Layout Selection for Pedestrian Localization and Its Acceleration Using GPU**
  *Woonhyun Nam, Bohyung Han, and Joon Hee Han*  
  *Computer Vision and Image Understanding (CVIU), 120: 46-58, 2014*

- **Canny Text Detector: Fast and Robust Scene Text Localization Algorithm**
  *Hojin Cho, Myungchul Sung, Bongjin Jun,*  
  *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), Las Vegas, USA, 2016 (to appear)*

- **Learning to Select Pre-trained Deep Representations with Bayesian Evidence Framework**
  *Yong-Deok Kim, Taewoong Jang, Bohyung Han, Seungjin Choi*  
  *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), Las Vegas, USA, 2016 (to appear)*

- **Scene Text Detection with Robust Character Candidate Extraction Method**
  *Myung-Chul Sung, Bongjin Jun, Hojin Cho, Daijin Kim,*  

*Plus 20+ papers @ major conference/journal from StradVision’s algorithm engineers @ POSTECH*
# Automotive Product Roadmap

<table>
<thead>
<tr>
<th>Platform</th>
<th>Features</th>
<th>Camera</th>
<th>2017</th>
<th>2018</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>1M 2M 3M 4M 5M 6M 3Q 4Q</td>
<td>1Q 2Q 3Q 4Q</td>
</tr>
<tr>
<td>High Seg</td>
<td>NVIDIA PX2</td>
<td>PD/VD, LD, FSD</td>
<td>Frontal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NVIDIA TX2</td>
<td>PD/VD, LD, FSD</td>
<td>Frontal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NVIDIA TX1</td>
<td>PD/VD</td>
<td>Frontal</td>
<td>March</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PD</td>
<td>Frontal</td>
<td>March</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PD</td>
<td>Frontal</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>VD</td>
<td>Side</td>
<td>March</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PD</td>
<td>Rear</td>
<td>March</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PD/VD</td>
<td>Rear</td>
<td>March</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PD/VD</td>
<td>AVM</td>
<td>March</td>
</tr>
<tr>
<td>Low Seg</td>
<td>ARM</td>
<td>PCW, FCW, PD, VD, LD</td>
<td>Frontal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ARM</td>
<td>PD</td>
<td>Frontal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ARM</td>
<td>PD/VD</td>
<td>Frontal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ARM</td>
<td>POD</td>
<td>Internal</td>
<td></td>
</tr>
<tr>
<td>Server</td>
<td>PC</td>
<td>PD/VD, Attributes</td>
<td>Frontal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PC</td>
<td>PD/VD, Attributes</td>
<td>Frontal</td>
<td></td>
</tr>
</tbody>
</table>

Left edge = First Prototype; Right edge = Second Prototype
Thanks for listening!

Any Questions / Comments, please contact contact@stradvision.com