Outdoor Augmented Reality with Geo-Object Identification based on Deep Learning Network

Prof Dr Du Qingyun; Rao Jinmeng; Qiao Yanjun
qydu@whu.edu.cn
01/ Background

Domains

• game, social activities, life, education...

Technologies

• Image Registration
  • Non-visual Perception
  • Visual Identification

Trend

• Indoor to Outdoor
• Wearable computing
• LBS & Augmented Reality
• Timeliness and Accuracy
Feature points - based

- Complicated computing
- Error-prone
- Timeliness not guaranteed
- Human involved

Deep Learning - based

- GPU acceleration needed
- High accuracy
- fast
- End-to-end black box computing
Development:
- CNN was firstly used in ImageNet competition in 2012
- Errors reduced by 10% to 16.4% compared to feature points based approach
- Continue to reduce till to 3.57%

Applications:
- Image Classification
- Object Identification
- Gesture Evaluation
- Image Segmentation
- Face Recognition

Mainstream Models:
- (Convolutional Neural network)
  - R-CNN, Fast R-CNN, Faster R-CNN, SSD, YOLO

Advantages:
- Highly reliable
- Timeliness
- End-to-end
- Training once, using everywhere
02
PART TWO
Roadmap
02 Roadmap

Visual Identification Model
- Simplified CNN

01 Mixed Tracing and Registration
- Sensor based non-visual context perception

02 LBS
- surrounding retrieval
- set limit to scope
- Attribute association

03 Augmented Information
- 2D attributes
- 3D Geometry
- HCI

04 3D Image registration
- Mapping from 3D object to 2D screen coordinate system
- Reversed coordinate system mapping
PART THREE

Methodology and Results
03 Methodology

A  Model Design
B  Model Training
C  Prototype system design
D  System Implementation
E  Execution and Test
Choosing a mode

- Based on Google research testing on mainstream CNN models, SSD (Single Shot Multibox Detector) has better performance on detection and identification of larger object.
- SSD was simplified in model structure to facilitate using on mobile platform with reasonable performance.

**SSD**

- Simplified SSD

**ResNet-18**

-经过Conv3_2，Conv4_2，Conv5_2
### 1. Model Design

#### Model parameters

<table>
<thead>
<tr>
<th>Convolutional Layer</th>
<th>Output size</th>
<th>Previous 18-layered network</th>
<th>Simplified network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conv1</td>
<td>112*112</td>
<td>7*7,64, stride:2</td>
<td>7*7,64, stride:2</td>
</tr>
<tr>
<td>Conv2_x</td>
<td>56*56</td>
<td>[3*3, 3,64]</td>
<td>3*3, 3,32</td>
</tr>
<tr>
<td>Conv3_x</td>
<td>28*28</td>
<td>[3*3, 3,128]</td>
<td>3*3, 3,32</td>
</tr>
<tr>
<td>Conv4_x</td>
<td>14*14</td>
<td>[3*3, 3,256]</td>
<td>3*3, 3,32</td>
</tr>
<tr>
<td>Conv5_x</td>
<td>7*7</td>
<td>[3*3, 3,512]</td>
<td>3*3, 3,32</td>
</tr>
</tbody>
</table>

#### Pre network

<table>
<thead>
<tr>
<th>Layer</th>
<th>Kernel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conv8_1</td>
<td>1<em>1</em>256, stride:1</td>
</tr>
<tr>
<td>Conv8_2</td>
<td>3<em>3</em>512, stride:2</td>
</tr>
<tr>
<td>Conv8_11</td>
<td>1<em>1</em>128, stride:1</td>
</tr>
<tr>
<td>Conv8_22</td>
<td>3<em>3</em>256, stride:2</td>
</tr>
</tbody>
</table>

#### Connected network

<table>
<thead>
<tr>
<th>Layer</th>
<th>Kernel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pool10</td>
<td>1*, aver. Pooling layer</td>
</tr>
</tbody>
</table>

#### Parameter of Default Box

<table>
<thead>
<tr>
<th>Layer</th>
<th>default box No</th>
<th>Default box Ratio</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conv3_2</td>
<td>3</td>
<td>[1,2,0.5]</td>
<td>In [1,2,0.5], each value</td>
</tr>
<tr>
<td>Conv4_2</td>
<td>5</td>
<td>[1,2,0.5,3,1/3]</td>
<td>ratio of length and width of</td>
</tr>
<tr>
<td>Conv5_2</td>
<td>5</td>
<td>[1,2,0.5,3,1/3]</td>
<td>default box</td>
</tr>
<tr>
<td>Conv8_2</td>
<td>5</td>
<td>[1,2,0.5,3,1/3]</td>
<td></td>
</tr>
<tr>
<td>Conv9_2</td>
<td>5</td>
<td>[1,2,0.5,3,1/3]</td>
<td></td>
</tr>
<tr>
<td>Pool10</td>
<td>5</td>
<td>[1,2,0.5,3,1/3]</td>
<td></td>
</tr>
</tbody>
</table>
03/ 2. Model Training

Data capture
- 3 Categories of object
- 300 images for each Category

Data Handling
- Mark the bounding box for objects and their category
- Generate XML configuration file

Training Environment
- Platform: MXNET
- Processor: GPU
- OS: Linux Ubuntu 14.04
- Language: python
Result

Model training lasted for 12 hours with 224 times of iteration to reach convergence. mAP to 58.2%. 0.03 second to finish the detection for one image. Model can be packaged into .so library for mobile usage.
03/ 3. Prototype System Design

General Framework

- Image input and output
- Info query and assoc.
- 2D & 3D Display

Image Handling

- Image Capture
- Image Transformation
- SSD Engine
- Detection result interpretation

Position and Gesture Computing

- Relative Position
- Camera Orientation
- Camera Gesture

Spatial Query

- POI
- SQLite Attribute Query
- Surrounding Query

Augmented Information

- Text Design
- Texture Design
- Graphics Design
- Coordinate Transformations
03 3. Prototype System Design

Image Processing Model

- Android Camera
- Capture On-site Image
- YUV to RGB
- Image Output
- Mobile Display
- Detection Information
- Convolutional Computing
- Extraction of Feature Layer
- Linear Regression of Bounding Box
- SoftMax Classification
- SSD Engine
03/ 3. Prototype System Design

3D Image Registration Algorithm

1. 根据视景体来计算X，Y轴的长度（刻度），
   \( top = Z \times \arctan(\frac{\theta}{2}) \)，根据屏幕的长宽比width/height来计算出
   right的值，right = top \times \text{width/height}。

2. 计算出(u,v)坐标的占比，
   \( u / \text{width}, v / \text{height} \)。

3. 计算(x,y)坐标系下的坐标，
   \( x = (u / \text{width}) \times 2 \times \text{right} - \text{right} \)，
   \( y = \text{top} - (v / \text{height}) \times 2 \times \text{top} \)。

4. 计算三维图形的四个坐标：定义宽度b_width
   为1和高度b_height为1。
### Augmented Info

#### 2D Attribute Info

<table>
<thead>
<tr>
<th>id</th>
<th>cn_name</th>
<th>ob-name</th>
<th>longitude</th>
<th>latitude</th>
<th>attribute</th>
<th>Picture</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SRES build.</td>
<td>SRES</td>
<td>30.522716</td>
<td>114.35506</td>
<td>text label</td>
<td>Binary</td>
</tr>
<tr>
<td>2</td>
<td>3S Statue</td>
<td>3SM</td>
<td>30.52798</td>
<td>114.355703</td>
<td>text label</td>
<td>Binary</td>
</tr>
<tr>
<td>3</td>
<td>Teaching build.</td>
<td>TEB</td>
<td>30.527728</td>
<td>114.356254</td>
<td>text label</td>
<td>Binary</td>
</tr>
</tbody>
</table>
03/ 3. Prototype System Design

Interface Design

SurfaceView

- Detection Box
- Category and Probability
- Real Object
- 3D Graphics and Texture

GLSurfaceView

- Image Preview
- Start
- Pause
Development Environment
Android, Version 4.4.2, SDK: 25, JDK: java7
Tool: Android Studio 2.2.3, Language: java, Database: SQLite3
### Testing Statistics

<table>
<thead>
<tr>
<th>Time</th>
<th>Category</th>
<th>Average confidence</th>
<th>Ave time lapse (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8:30-9:00</td>
<td>SRES Build.</td>
<td>0.89</td>
<td>1486</td>
</tr>
<tr>
<td></td>
<td>3S Statue</td>
<td>0.87</td>
<td>1438</td>
</tr>
<tr>
<td></td>
<td>Teaching Build.</td>
<td>0.91</td>
<td>1509</td>
</tr>
<tr>
<td>11:30-12:00</td>
<td>SRES Build.</td>
<td>0.92</td>
<td>1502</td>
</tr>
<tr>
<td></td>
<td>3S Statue</td>
<td>0.91</td>
<td>1549</td>
</tr>
<tr>
<td></td>
<td>Teaching Build.</td>
<td>0.93</td>
<td>1527</td>
</tr>
<tr>
<td>17:30-18:00</td>
<td>SRES Build.</td>
<td>0.91</td>
<td>1456</td>
</tr>
<tr>
<td></td>
<td>3S Statue</td>
<td>0.86</td>
<td>1523</td>
</tr>
<tr>
<td></td>
<td>Teaching Build.</td>
<td>0.92</td>
<td>1501</td>
</tr>
</tbody>
</table>
Testing with two objects

System execution

HCI
4. System Implementation
PART SIX
Outlook
04 What was achieved

01 Introducing spatial information into mixed image tracing and registration
Identify the category of object using both visual info and non-visual spatial info, by which searching scope is much smaller by surrounding query.

02 Using deep learning in outdoor augmented reality
Based on SSD model, by modifying pre-network, network structure was simplified. Geographic object identification can be 90% accurate and less than 1.5s is needed for one image.

03 Implementing Geo-AR system on mobile phone
Localize SSD engine on Android platform. Integrating with sensor system of mobile phone, using local SQLite database, 3D registration was realized with algorithms of 2D-3D transformation.
04 Outlook

- Structure optimization of CNN
- Tracing Accuracy Improvement
- Augmented Information Enrichment
Thank you for attention!