A Review of Intelligent Repair Processes for End-of-Life Components

Authors: Eman Shayeb
Ms. Habiba Imam
Ms. Rabiya Abbasi
Dr. Rafiq Ahmad *
An Introduction to Computer Vision

What is Computer Vision?
• Computer vision is a field related to but separate than AI that aims to create machines that are able to perceive the world in the same way that humans do. ¹
• ConvNets are important because they currently serve as the basis of development for computer vision ²

Where Can I Find Computer Vision In my Daily Life? ³
• Facial Recognition
• Image classification
• Recommendation Systems
• Natural Language Processing

Convolutional Neural Networks
• ConvNets are Deep Learning Algorithms that can differentiate between parts of an image by attaching weights/biases (significance) to certain parts of said image. ⁴
  – The ConvNet is built similarly to the visual cortex of the human brain because of its connectivity and behaviors in the Receptive Field. ⁵
  – ConvNets can be trained to recognize features on their own. ⁶

2 Ibid
3 Ibid
4 Ibid
This presentation serves as an exploration of the different kinds of detection models, and their strengths and weaknesses. It will also include applications and functions of said deep learning detection models.

It also serves as justification for the use of Faster R-CNN rather than other architectures for damage detection.

**Project**

This computer vision project uses the Faster R-CNN architecture for damage detection.

Main goal: train Faster R-CNN architecture to detect damage on a cylindrical shape. Damage will be repaired, and machine will be able to automatically stop when cylinder is fully repaired.

This can be applied to many areas such as detection of different types of damage on different types of objects:

- Damage detection and repair on pipes (water pipes, oil pipes etc.)
- Damage detection on cylindrical building pillars
- Other mechanical components

The Architecture is currently trained to detect prominent areas of damage, but Faster R-CNN can be trained to smaller types of corrosion, cracks, and dents
Research Motivation and Methodology

Motivation
Damage detection architectures can be applied in many different ways to improve daily life and manufacturing projects. One of the most important factors is being able to identify which architectures and programs will allow for the most efficient damage detection.

Methodology
To gather the information for this particular project, the keywords used were “damage detection”, “computer vision”, “convolutional neural networks”.

Due to the virtual nature of this research, digital libraries and databases were the main source of information for it. This is also the reason why my own is largely absent from this project.
The emergence of Convolutional Network Based Detection

CNN’s were quite popular in the 1990’s, until they fell off during the early 2000s because of the emergence of support vector machines.

Krizhevsky et al. released a paper in 2012 that demonstrated CNN’s higher image classification accuracy, thus reviving interest in the use of CNN’s.

Traditional Object Detection Models

consist of 3 stages:

**Informative Region Selection**
- +scans whole image with multi-scale sliding window.
- - is computationally expensive
- - too many redundant windows

**Feature Extraction**
- +extracting features provides robust representation
- - different backgrounds and illumination mean that creating a descriptor that applies to *all* objects is difficult

**Classification**
- +separates object from surrounding area

---

**What Makes CNN’s better than MLP’s?**

<table>
<thead>
<tr>
<th>Convolutional Neural Networks (CNN)</th>
<th>Multilayer Perceptron (MLP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Linear transformation invariant</td>
<td>• Not Linear transformation invariant</td>
</tr>
<tr>
<td>• depth is created (nearby pixels are more strongly related than far ones)</td>
<td>• Loss of spatial information, this means that images would not be able to be defined as clearly</td>
</tr>
<tr>
<td>• usage of filters allows for feature identification</td>
<td>• Many weights and biases that make MLP’s more computationally heavy than CNN’s</td>
</tr>
<tr>
<td>• It is very unlikely for the CNN to learn the same feature because filters are random numbers that continuously update as the CNN is trained. (unless # of filters is really large)</td>
<td></td>
</tr>
</tbody>
</table>

CNN Architecture

**Typical CNNs have...**

- **Receptive field** -- each neuron is connected to neurons of adjacent layers.
- **Layers called feature maps**
- **Input layer** -- 3D matrix of pixel intensities for different colour channels
- **Internal layer** -- multichannel image

**Terms:**

- **Convolution layer** - creates a prediction of each feature of the image through the creation of a feature map. This is done through a filter that scans the image, pixel by pixel.

- **Pooling layer** – After the conv layer creates the information from the image, the pooling layer will keep the necessary information.

- **Fully Connected Input Layer** – flattens previous outputs to form a single vector, which will be input into the next layer.

- **Fully connected output layer** – Final touches; determines final probabilities for classifying an image.

---


**R-CNN**

This was an object detection model introduced in a paper by Ross Girshick et al. in 2014.

+ The model essentially cuts the image into pieces for more efficient detection/localization. (segmentation)
- 2000 region proposals per image
- takes about 49 seconds to detect a single image
- computationally expensive and uses up a lot of disk space

**Fast R-CNN**

This is an improved object detection model introduced in a paper by Ross Girshick et al. in 2015

+ All region proposals are input together as one forward propagation
+ combines things like ConvNet and classification layers which saves disk space by avoiding the need to save
+ training/detection time are decreased from R-CNN counter part
  - from 84 hours to 8.75 hours
- selective search takes up most of the computation time

---


Faster R-CNN detection process:
1. Insert image.
2. Image passes through a trained CNN to create a feature map.
3. Region Proposal Network will use the features computed (in step 2) to create bounding boxes.
4. Create list of probabilities for potential objects + their location in image.
5. Apply Region of Interest Pooling – this picks out features that relate to (relevant) objects.
6. R-CNN kicks in and classifies objects in bounding boxes.

Many detection architectures have been based on Faster R-CNN, and other architectures could arguably have more “advanced” features, but that is not necessary for this particular project because the damage is usually large without much variance -- making it easier for the system to detect.

• Example 1: **MASK R-CNN** is based on Faster R-CNN, but it adds onto the pre-existing branch for bounding boxes with branch for instance/semantic segmentation **which is not required for this project.**

### Analysis

<table>
<thead>
<tr>
<th>REAL TIME IMAGE PROCESSING</th>
<th>DAMAGE QUANTIFICATION</th>
<th>DAMAGE DETECTION</th>
<th>DAMAGE IDENTIFICATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-CNN</td>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
<tr>
<td>FAST R-CNN</td>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
<tr>
<td>FASTER R-CNN</td>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
<tr>
<td>MASK R-CNN</td>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
<tr>
<td>YOLO</td>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
<tr>
<td>SSD</td>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
<tr>
<td>R-FCN</td>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
</tbody>
</table>

**Key:**
- ✨: minimal issues
- 🟠: runs into some issues
- 🔴: runs into many issues

<table>
<thead>
<tr>
<th>CORROSION</th>
<th>CRACK</th>
<th>DENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
<tr>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
<tr>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
<tr>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
<tr>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
<tr>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
<tr>
<td>✨</td>
<td>🟠</td>
<td>🟠</td>
</tr>
</tbody>
</table>

### References

- [https://www.forskningsdatabasen.dk/en/catalog/2389380154](https://www.forskningsdatabasen.dk/en/catalog/2389380154)
- [https://www.researchgate.net/profile/Young-Jin_Cha/publication/321341582_Autonomous_Structural_Visual_Inspection_Using_Shifted_Dielectric_Frequency_Reflectometry/links/5a1dbbaca6fdccc6b7f848d0/Autonomous_Structural_Visual_Inspection_Using_Shifted_Dielectric_Frequency_Reflectometry.pdf](https://www.researchgate.net/profile/Young-Jin_Cha/publication/321341582_Autonomous_Structural_Visual_Inspection_Using_Shifted_Dielectric_Frequency_Reflectometry/links/5a1dbbaca6fdccc6b7f848d0/Autonomous_Structural_Visual_Inspection_Using_Shifted_Dielectric_Frequency_Reflectometry.pdf)
- [https://ieeexplore.ieee.org/abstract/document/8937176/keywords#keywords](https://ieeexplore.ieee.org/abstract/document/8937176/keywords#keywords)
- [https://search-proquest-com.login.ezproxy.library.ualberta.ca/docview/2386649388?accountid=14474](https://search-proquest-com.login.ezproxy.library.ualberta.ca/docview/2386649388?accountid=14474)

---

**Dr Rafiq Ahmad**
https://sites.ualberta.ca/~rafiq1/

**Eman Shayeb**
Realemanshayeb@gmail.com

**WISEST**
July 29, 2020
A study done by Jonathan Huang et al. in 2017 compares multiple trade-offs for 3 main convolutional object detectors, Faster R-CNN, R-FCN, and SSD. It is important to note that SSD and R-FCN are based on Faster R-CNN.

Sources

<table>
<thead>
<tr>
<th></th>
<th>Faster R-CNN</th>
<th>R-FCN</th>
<th>SSD</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sensitivity to feature extraction quality</strong></td>
<td>More sensitive than SSD</td>
<td>More sensitive than SSD</td>
<td>Less sensitive than Faster R-CNN and R-FCN</td>
</tr>
<tr>
<td><strong>Speed and accuracy</strong></td>
<td>Slower than the other two, but more accurate</td>
<td>Faster but less accurate than Faster R-CNN</td>
<td>Faster but less accurate than Faster R-CNN</td>
</tr>
<tr>
<td><strong>Object Size</strong></td>
<td>Scales according to size of object, computes more pixels</td>
<td>Scales according to size of object, computes more pixels</td>
<td>Resizes images, computes less pixels</td>
</tr>
<tr>
<td><strong>Small Object</strong></td>
<td>Performs well</td>
<td>Performs well</td>
<td>Performs very poorly</td>
</tr>
<tr>
<td><strong>Large Object</strong></td>
<td>Performs well</td>
<td>Performs well</td>
<td>Performs well</td>
</tr>
<tr>
<td><strong>Image Resolution</strong></td>
<td>Performance accuracy decreases in low resolution, and vice versa</td>
<td>Performance accuracy decreases in low resolution, and vice versa</td>
<td>Performance accuracy decreases in low resolution, and vice versa</td>
</tr>
<tr>
<td><strong>Proposal Number</strong></td>
<td>Less proposals will speed it up and maintain accuracy</td>
<td>Less proposals will speed it up and maintain accuracy</td>
<td>Less proposals will speed it up and maintain accuracy</td>
</tr>
</tbody>
</table>
## Other Real Life Applications

### Traffic Sign Detection (Faster R-CNN)
- Helps with unmanned/automatic driving
- A study done by Faming Shao et al. uses faster R-CNN to reduce computing time for traffic sign detection


### Smart Nursery (Faster R-CNN)
- Somnuk Phon-Amnuaisuk et al. explored the development of a Smart Nursery with Faster R-CNN
- This includes detecting the baby and what the baby is doing (sleeping, crawling, playing etc.)

---


### Vehicle Detection (YOLO and SSD)
- Mehdi Masmoudi et al. explored various detection models that were able to detect vehicles in real time
- Their findings can be used for increasing the efficiency vehicle/object detection in fully automated driving


### Aerial Vehicle Imagery (Faster R-CNN, R-FCN, SSD, and YOLO)
- Suk-Ju Hong et al. published a study that used multiple detection models to track and count birds in their natural habitats
- Their development of these models can be applied to future aerial imagery with varying resolutions and flight altitudes.

Common Issues of Computer Vision

- Problems with object localization, detection, recognition
  - Transformation variants
  - Shape deformation variants
  - Lighting and colour variants

This includes variations in scale / translation / rotation.

Conclusion

- CNN’s form the basis of modern computer vision technologies and regained popularity in 2012.
- Traditional Object detection models, like MLP’s, have 3 distinct stages: Region Selection, Feature Extraction, Classification.
- R-CNN → combine ConvNet and classification layers to save disk space → Fast R-CNN → adjusting selective search method → Faster R-CNN.
- Faster R-CNN is used for this particular project because of its speed and accuracy, but also relative simplicity compared to models like MASK R-CNN.
- These models can be applied in many areas like traffic detection for unmanned driving to monitoring animals to creating smart nurseries.
- There tend to be 3 main variants that must be overcome by computer vision: Transformation, shape, lighting/colour.

I really enjoyed doing research on computer vision this Summer! I will definitely continue exploring this field in the future, and I am excited to see what new technologies develop over time!
Acknowledgements

• I would like to thank my sponsors, the Edmonton Chapter of Beta Sigma Phi and Motorola – without them I would not have been able to participate in the program in the first place.

• I would also like to thank WISEST for giving me this once in a lifetime opportunity to participate in such a unique program, and for adapting so seamlessly to the virtual program (it almost feels like I was on campus).

• I would additionally like to thank the LIMDA team for helping me throughout my research – particularly Dr Rafiq Ahmad, Ms. Habiba Iman, and Ms. Rabiya Abbasi for supervising and aiding me throughout this process!

• Thank you for a wonderful Summer of making connections and learning a ton!
References


Any Questions?

Eman Shayeb
Realemanshayeb@gmail.com