Crack Detection in Concrete Surfaces using Image Processing, Fuzzy Logic, and Neural Networks

Gajanan K. Choudhary and Sayan Dey

Department of Civil Engineering
Indian Institute of Technology (IIT)
Kharagpur, India

Introduction: Need

- Structural health monitoring (SHM):
  - Skilled labor
  - Time consuming
  - Costly
- SHM in inaccessible or dangerous areas
  - Sewers
  - Nuclear reactors

Introduction: Previous Work

- Initial work focus solely on image processing.
- Kaseko *et al*.: Neural network
- Chae *et al.*: Neuro-fuzzy system
- Fujita *et al.*: Probabilistic relaxation, local thresholding, and 3 measures of accuracy - sensitivity, specificity and precision
- Sinha *et al.*: Neuro-fuzzy model, *type* of crack
- Moon and Kim: Neural network with 5 hidden layers with image-wise accuracy as measure

Objectives

- Comparative study of different types of approaches to crack detection
- Studying various measures of model performance
- Investigate the effect of output thresholds on performance measures
- Novel model based purely on fuzzy logic, unlike previous neuro-fuzzy approaches
- Comparative study of effect of size of neural networks on performance measures
- Comparative study of edge detection, median filter, Gaussian filter, etc. on performance measures

Major Steps

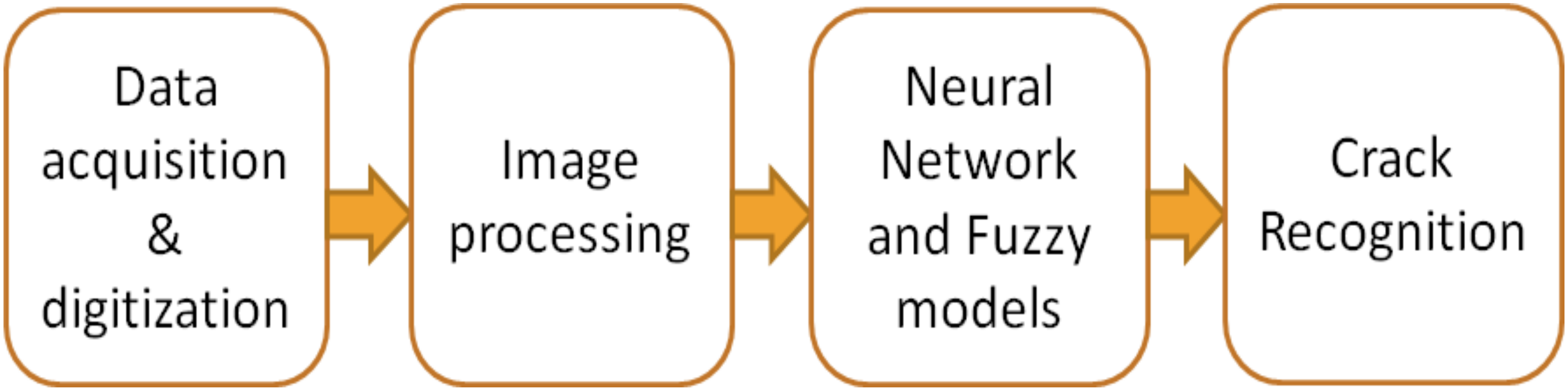

Data acquisition & digitization
Image processing
Neural Network and Fuzzy models
Crack Recognition

Image Processing

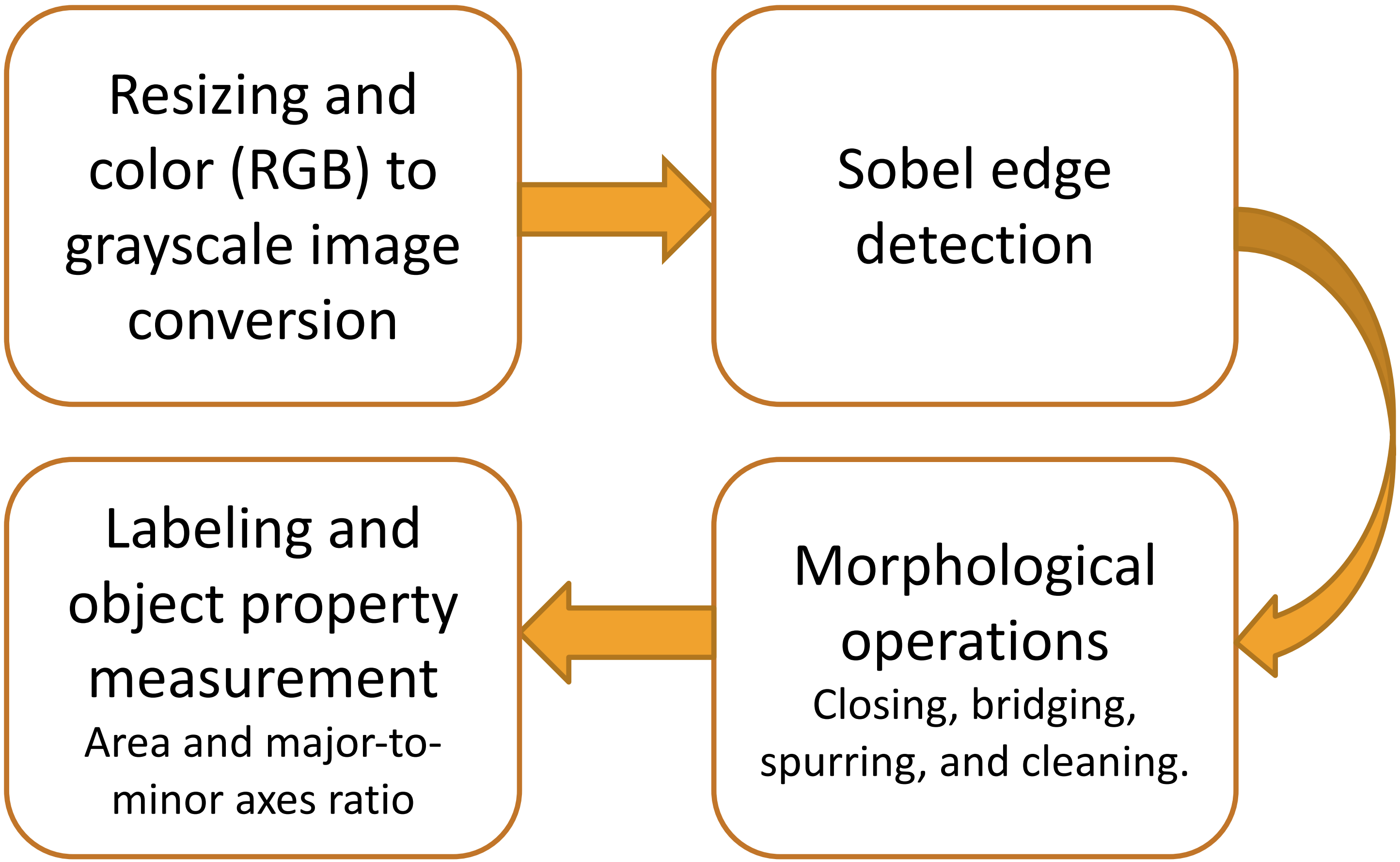
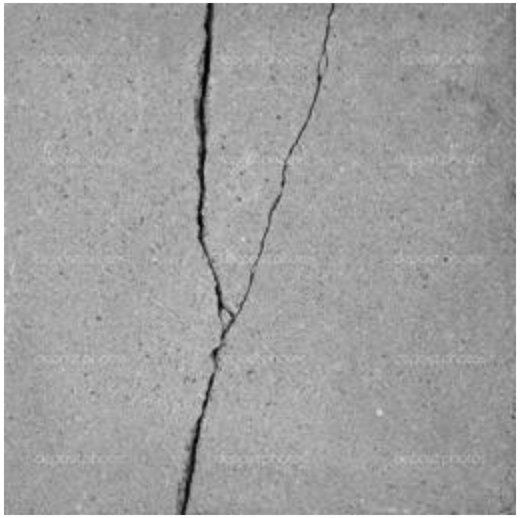
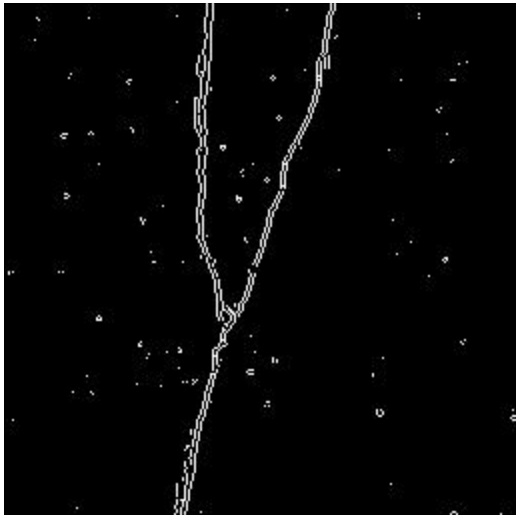
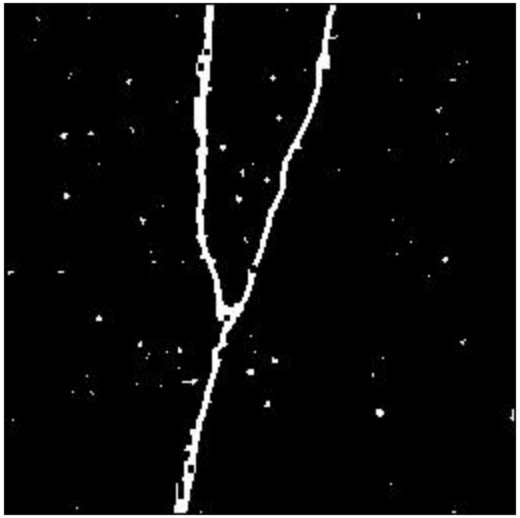
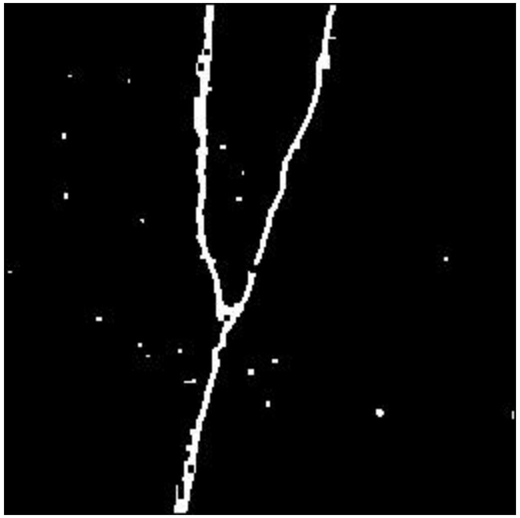


Image Processing: Example









Base Image Source:

http://static5.depositphotos.com/1013817/399/i/950/depositphotos_3992892-Elegant-split-crack-in-concrete.jpg

Approaches Toward Model Development

- Object approach
- Image approach
- Threshold fine-tuning
- Performance measures:
  - Sensitivity
  - Specificity
  - Precision
  - Image-wise accuracy
  - Object-wise accuracy

Fuzzy Logic Model (Object Approach)

- Mamdani fuzzy inference system
- Membership functions:
  - Input Variables
    - Area – low, moderate, and high
    - Major-to-minor axes ratio – low and high
  - Output Variable:
    - Class – Noise and crack
- ‘Min’ implication for ‘AND’ operator
- ‘Max’ aggregation with ‘centroid’ defuzzification

Neural Network Approaches

- Image approach architectures:
  - 100 – 50 – 50 – 50 – 1
  - 100 – 10 – 10 – 10 – 10 – 1
  - 100 – 5 – 5 – 5 – 5 – 5 – 1
  - 100 – 5 – 5 – 5 – 5 – 5 – 5 – 1
- Object approach architectures:
  - 2 – μ – 1, where 20 ≤ μ ≤ 1
- Single neuron output between 0 and 1
- Threshold value $0 < \alpha < 1$ must be chosen to differentiate between cracks and noise

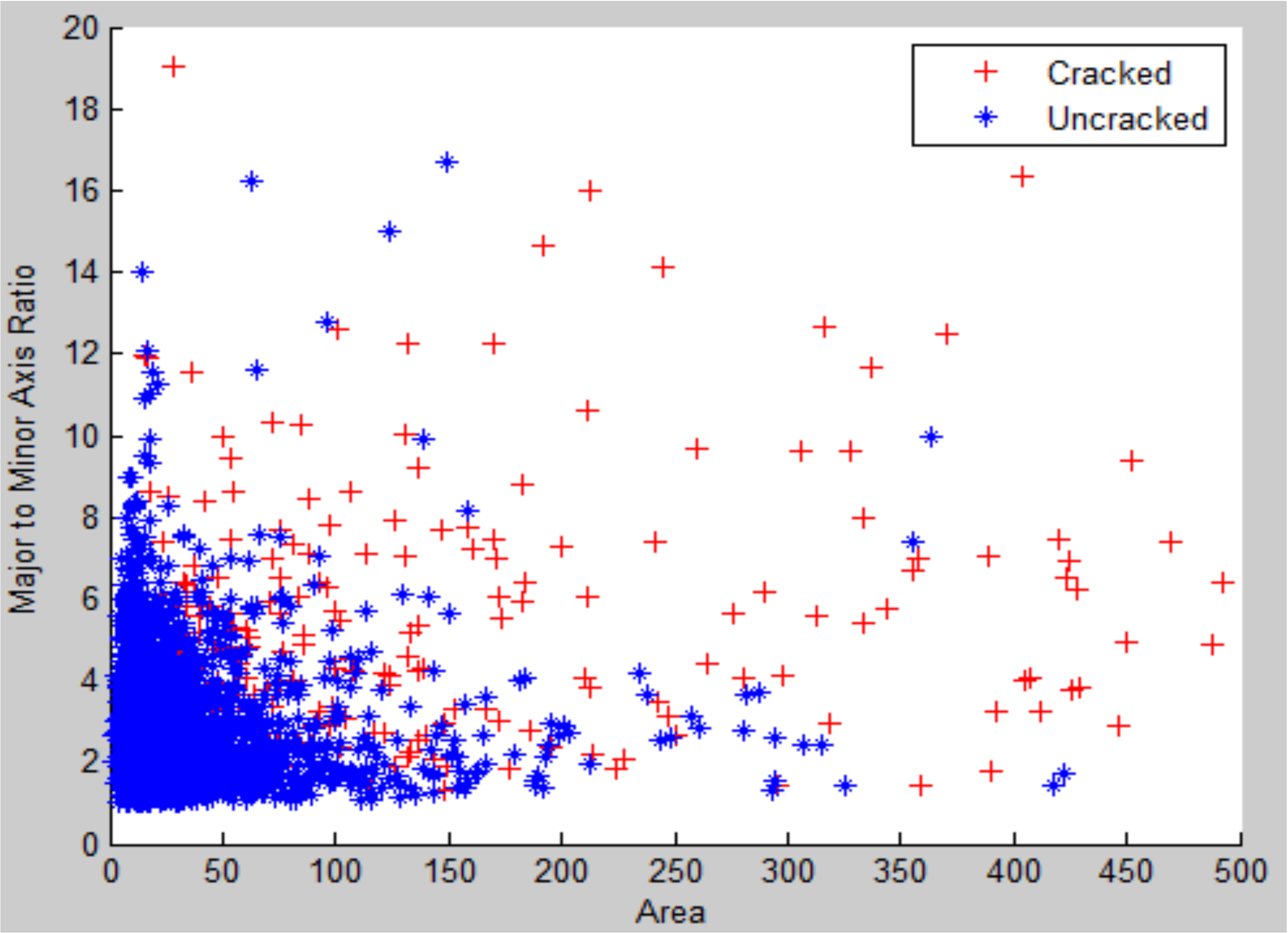
Image Approach: Network Properties

- Feed forward back propagation
- Training function: Scaled conjugate gradient
- Transfer function: Log-sigmoid function
- Learning rate: 0.01
- Momentum: 0.9
- Maximum epochs: 10000
- Training: 70% of samples.
- Testing: 15% of samples.
- Validation: 15% of samples.

Image Approach: Network Properties

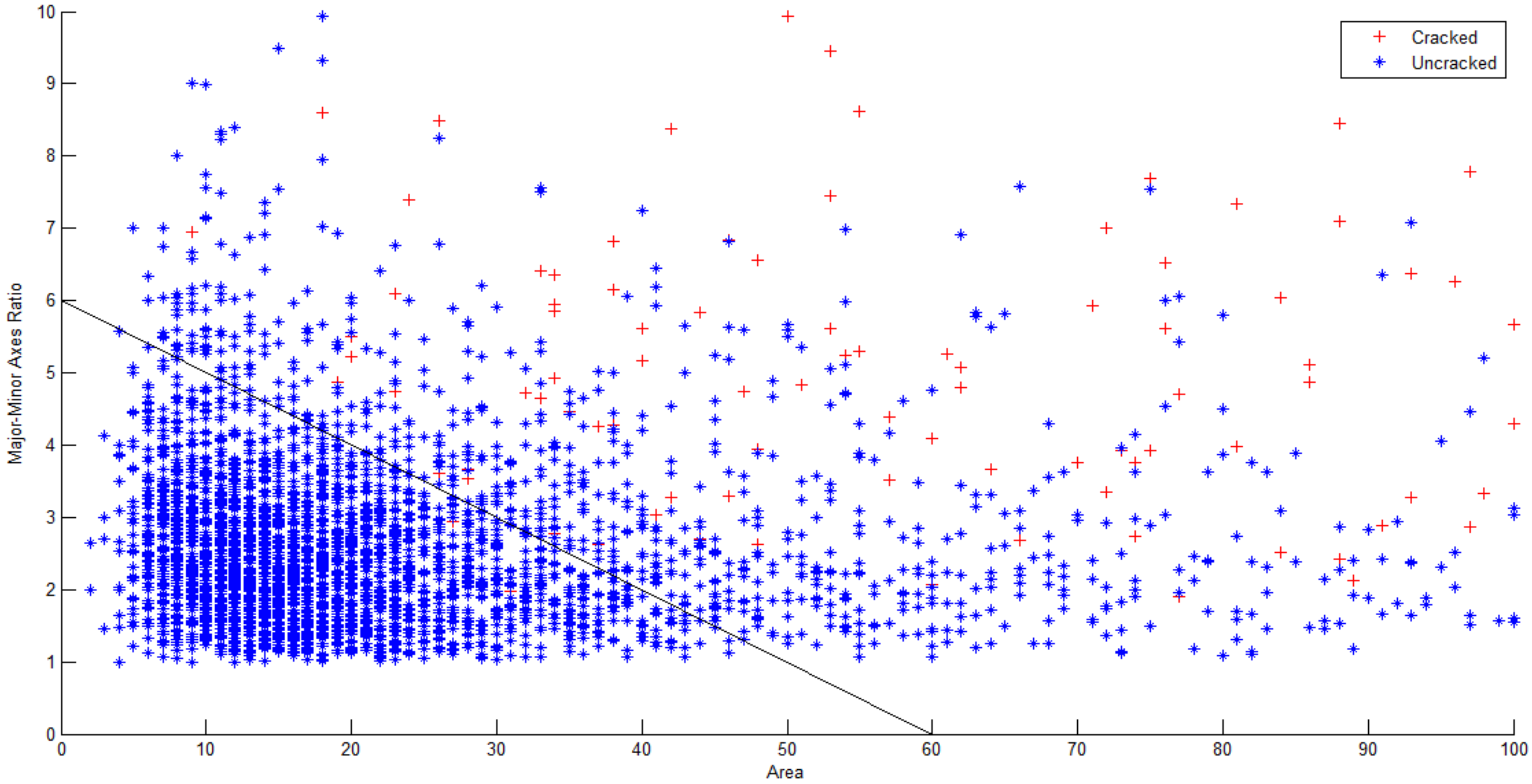
- Input: 100 neurons
  - Area of 50 objects per image (in descending order)
  - Major-to-minor axes ratios corresponding to (descending) area of each object per image
- 3 to 6 hidden layers
- Equal number of hidden neurons - multiples of 5
- Output: 1 neuron
  - 1 – Crack
  - 0 – Noise

OBJECT APPROACH: 2-CLASS CLASSIFICATION



Object Approach: Imbalance in Dataset

- 12000 noise : 300 cracks reduced to 400 noise : 300 cracks
- 70% of noise above line, 7.5% of noise below.



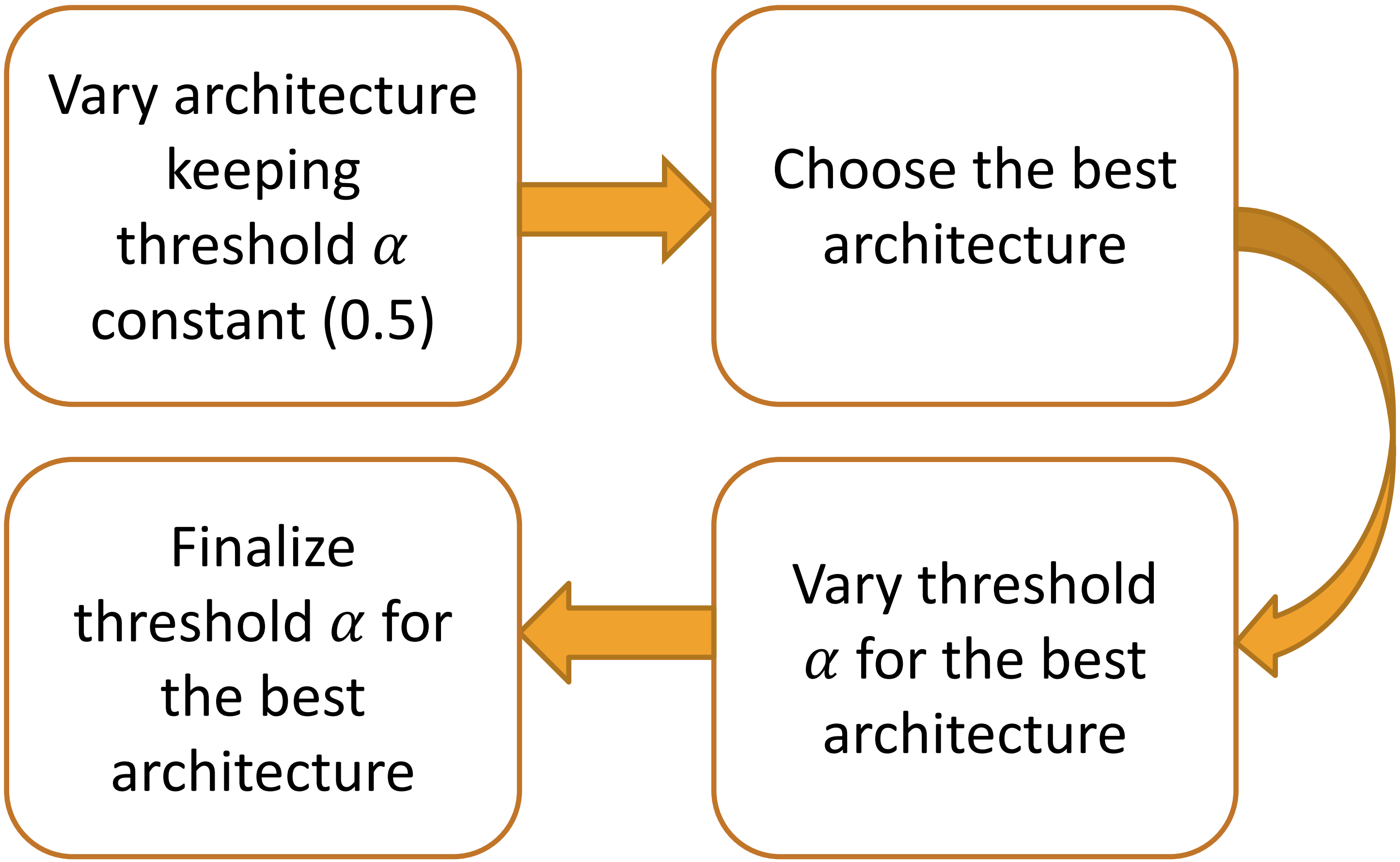
Object Approach: Network Properties

- Feed forward back propagation
- Training function: Scaled conjugate gradient
- Transfer function: Log-Sigmoid activation
- Maximum epochs: 10000
- Training: 70% of samples
- Testing: 15% of samples
- Validation: 15% of samples

Object Approach: Network Properties

- Input: 2 neurons
  - Area of an object
  - Major-to-minor axes ratio of that object
- Single hidden layer with 1 to 20 neurons
- Output: 1 neuron
  - 1 – Crack
  - 0 – Noise

OBJECT APPROACH: FINALIZING MODELS



Results: Measures

- $\text{Image} - \text{wise accuracy} = \dfrac{\text{Images correctly classified}}{\text{Total number of images}}$
- $\text{Object} - \text{wise accuracy} = \dfrac{\text{Objects correctly classified}}{\text{Total number of objects}}$
- $\text{Sensitivity} = \dfrac{\text{Cracks correctly classified}}{\text{Total cracks}}$
- $\text{Specificity} = \dfrac{\text{Noise correctly classified}}{\text{Total noise}}$
- $\text{Precision} = \dfrac{\text{Cracks correctly classified}}{\text{Total objects classified as cracks}}$

Architecture Selection: Object Approach

No. of Hidden Nodes	Sensitivity	Specificity	Precision	Image-wise Accuracy	Object-wise Accuracy
1	0.1923	0.9997	0.9259	0.8667	0.9835
2	0.2692	0.9991	0.8642	0.8952	0.9845
3	0.5692	0.9959	0.7400	0.8286	0.9874
4	0.5731	0.9960	0.7450	0.8286	0.9875
5	0.5615	0.9966	0.7725	0.8571	0.9879
6	0.5731	0.9966	0.7760	0.8476	0.9881
7	0.5615	0.9965	0.7684	0.8667	0.9878
8	0.5808	0.9965	0.7704	0.8571	0.9881
9	0.5654	0.9962	0.7538	0.8286	0.9876
10	0.5577	0.9974	0.8146	0.8952	0.9886
11	0.5731	0.9965	0.7680	0.8571	0.9880
12	0.5577	0.9970	0.7923	0.8762	0.9882
13	**0.5808**	**0.9966**	**0.7784**	**0.8667**	**0.9883**
14	0.5769	0.9964	0.7653	0.8667	0.9880
15	0.5731	0.9969	0.7884	0.8762	0.9884
16	0.5769	0.9965	0.7692	0.8571	0.9881
17	0.5615	0.9966	0.7725	0.8667	0.9879
18	0.5654	0.9969	0.7903	0.8571	0.9883
19	0.5731	0.9964	0.7641	0.8571	0.9879
20	0.5615	0.9969	0.7892	0.8762	0.9882

Threshold For NN Object Approach

Threshold	Sensitivity	Specificity	Precision	Image-wise Accuracy	Object-wise Accuracy
0.65	0.5462	0.9975	0.8161	0.8857	0.9884
0.66	0.5462	0.9975	0.8161	0.8857	0.9884
0.67	0.5423	0.9975	0.8150	0.8857	0.9884
0.68	0.5385	0.9977	0.8284	0.8952	0.9885
0.69	0.5308	0.9980	0.8466	0.9048	0.9887
0.70	0.5269	0.9980	0.8457	0.9048	0.9886
0.71	**0.5269**	**0.9981**	**0.8509**	**0.9143**	**0.9887**
0.72	0.5231	0.9982	0.8553	0.9238	0.9887
0.73	0.5115	0.9983	0.8581	0.9238	0.9885
0.74	0.5077	0.9983	0.8571	0.9238	0.9884
0.75	0.5000	0.9983	0.8609	0.9238	0.9884
0.76	0.4962	0.9984	0.8658	0.9238	0.9884
0.77	0.4885	0.9984	0.8639	0.9238	0.9882
0.78	0.4885	0.9985	0.8699	0.9238	0.9883
0.79	0.4885	0.9987	0.8819	0.9333	0.9884
0.80	0.4808	0.9987	0.8865	0.9429	0.9884
0.81	0.4808	0.9988	0.8929	0.9524	0.9884
0.82	0.4692	0.9988	0.8905	0.9524	0.9882
0.83	**0.4538**	**0.9990**	**0.9008**	**0.9619**	**0.9881**
0.84	0.4462	0.9991	0.9063	0.9619	0.9880
0.85	0.4423	0.9991	0.9055	0.9524	0.9879

Comparison with Previous Work

Comparison with Fujita et al. [8]:					
Threshold	Sensitivity	Specificity	Precision	Image-wise Accuracy	Object-wise Accuracy
Sample threshold for the proposed model:					
0.25	0.792	0.989	0.592	0.733	0.985
Best results reported by Fujita et al. [8]:					
-	0.722	0.993	0.655	Global Thresholding	
-	0.815	0.922	0.147	Local Thresholding	
-	0.805	0.992	0.631	Probabilistic Relaxation	

Comparison with Previous Work

Comparison with Moon et al. [11]:			
Approach Type	**Models**	**Threshold α**	**Image-wise Accuracy**
Image	3 Hidden Layer NN	0.32	87.00
Image	4 Hidden Layer NN	0.41	86.00
Image	5 Hidden Layer NN	0.44	85.00
Image	6 Hidden Layer NN	0.70	93.00
Object	‘2-13-1’ NN	0.83	96.00
Object	Fuzzy Logic Model	0.68	94.00
Image	NN of Moon et al. [11]	-	90.25

Discussion: Accuracy

- Inverse proportionality:
  - Sensitivity and Specificity
  - Sensitivity and Precision
  - Sensitivity and Image-wise accuracy (below the optimum threshold)
- At threshold = 0 or 1, sensitivity = 1 or 0 and specificity = 0 or 1, respectively.
- *Maxima* of all measures attained at *different* thresholds

Discussion: Image Processing

- Image Processing
  - Fragmentation of cracks: Reason for low sensitivity
  - Sobel edge detection: Shadows, color changes as edge
  - Consider R,G and B separately
- Object Parameters
  - Sensitivity of all models low
  - Area and ratio alone not sufficient: Need additional parameters

Conclusion

- All 5 measures of performance important
- Object approach better than Image approach for all models
- Neural network models better than fuzzy logic model in all measures of performance, especially sensitivity
- For image approach, edge detection comparable to Gaussian filter/median filter etc.
- Image processing needs improvement

References

1. S. Tsao, N. Kehtarnavaz, P. Chan, and R. Lytton, "Image-based expert-system approach to distress detection on CRC pavement," *Journal of Transportation Engineering*, vol. 120, no. 1, pp. 62–64, January - February 1994.
2. K. C. Wang, S. Nallamothu, and R. P. Elliott, "Classification of pavement surface distress with an embedded neural net chip," *Artificial Neural Networks for Civil Engineers: Advanced Features and Applications*, pp. 131–161, January-February 1998.
3. M. S. Kaseko, Z. P. Lo, and S. G. Ritchie, "Comparison of traditional and neural classifiers for pavement-crack detection," *Journal of Transportation Engineering*, vol. 120, no. 4, pp. 552–569, July - August 1994.

References

4. Chae M., and Abraham D., (2001). ”Neuro-Fuzzy Approaches for Sanitary Sewer Pipeline Condition Assessment.” *Journal of Computing in Civil Engineering 15, Special issue: Information technology for life-cycle infrastructure management*, Pages 4–14. doi: 10.1061/(ASCE)0887-3801(2001)15:1(4).
5. Aws Khanfar, Mohammed Abu-Khousa, and Nasser Qaddoumi, “Microwave near-field non-destructive detection and characterization of disbonds in concrete structures using fuzzy logic techniques,” *Composite Structures*, Volume 62, Issues 3–4, Pages 335-339, ISSN 0263-8223, 10.1016/j.compstruct.2003.09.033, 2003.
6. Byoung Jik Lee, and Hosin “David” Lee, “Position-Invariant Neural Network for Digital Pavement Crack Analysis,” *Computer-Aided Civil and Infrastructure Engineering,* Volume 19, Issue 2, pages 105–118, March 2004.

References

7. Y. Fujita, Y. Mitani and Y. Hamamoto, "A Method for Crack detection on a Concrete Structure," *18th International Conference on Pattern Recognition*, Volume 3, pp. 901–904, 2006.
8. Y. Fujita, and Y. Hamamoto, "A robust method for automatically detecting cracks on noisy concrete surfaces," *Next-Generation Applied Intelligence. 22nd International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems IEA/AIE 2009*, pp. 76–85, June 2009, Tainan, Taiwan.
9. S. K. Sinha, P. W. Fieguth, and M. A. Polak, "Computer vision techniques for automatic structural assessment of underground pipes," *Computer-Aided Civil and Infrastructure Engineering*, vol. 18, no. 2, pp. 95–112, February 2003.

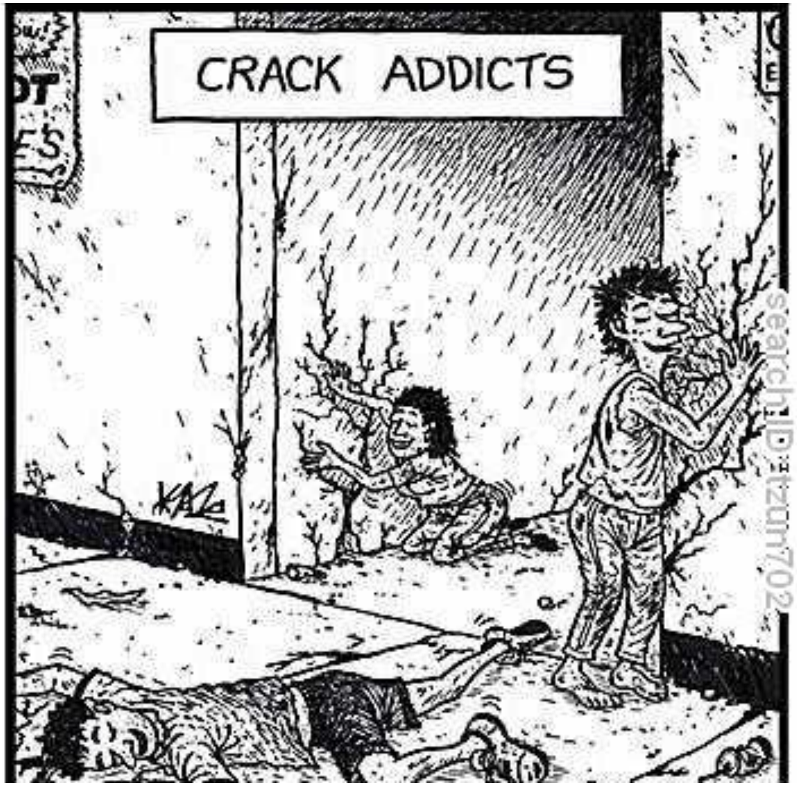
References

10. T. Yamaguchi, and S. Hashimoto, "Improved percolation-based method for crack detection in concrete surface images," *19^{th} International Conference on Pattern Recognition*, December 2008.
11. H. Moon, and J. Kim, "Inteligent Crack Detecting Algorithm On The Concrete Crack Image Using Neural network," *Proceedings of the 28^{th} ISARC*, Pages 1461-1467, Seoul, Korea, 2011.
12. The MathWorks, Inc., *Image Processing Toolbox for Use with MATLAB: User's Guide*, August 2004.
13. J. M. Twomey and A. E. Smith, "Validation and Verification," *19^{th} Artificial Neural Networks for Civil Engineers: Fundamentals and Applications*, Chapter 4. January 2005.

REFERENCES

14. L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, pp. 338-353, 1965.
15. J. -S. R. Jang and N. Gulley, *MATLAB Fuzzy Logic Toolbox: User's Guide*, April 1997.
16. E. H. Mamdani, "Applications of Fuzzy Set Theory to Control Systems: A Survey," in *Fuzzy Automata and Decision Processes*, M. M. Gupta, G. N. Saridis and B. R. Gaines, eds., North-Holland, New York, pp. 1-13, 1977.
17. M. H. Beale, M. T. Hagan and H. B. Demuth, *Neural Network Toolbox™: User's Guide,* September 2012.

- Concrete Structures Laboratory, Indian Institute of Technology Kharagpur, for images of concrete.
- Google Inc.: http://www.google.com - for images of concrete



Source: https://www.cartoonstock.com/cartoonview.asp?catref=tzun702

QUESTIONS?