Original Article

Object Detection And Training Of Deep Neural Networks

Deepika Yadav[1], **Omprakash Dewangan**[2]

[1]M.Tech. (Computer Science), Kalinga University, Raipur, Chhattisgarh, India
dipikayadav638@gmail.com
[2]Assistant Professor, Faculty of Information Technology, Kalinga University, Raipur, Chhattisgarh, India
omprakash.dewangan@kalingauniversity.ac.in
DOI: 10.47750/pnr.2023.14.S02.263

Abstract

Humans always want their daily basis tasks to be done without any intervention. Everything around us is filled with tremendous amount of perceivable information. As the technology advances, the amount of knowledge that we can obtain from that information also increases. And so, computer vision and artificial intelligence were introduced. Computer vision and artificial intelligence are one among the busy fields of technology in which advancements are constantly being introduced. Computer vision is a branch of science of computer systems which can recognize as well as understand images. Detecting and recognizing objects in unstructured as well as structured environments is one of the most challenging tasks in computer vision and artificial intelligence research and is one of the aspects of computer vision. We can use the science of object recognition for many useful applications in order to enhance the knowledge gained from the visible information around us.

This paper presents a trainable architecture of neural networks that can detect as well as recognize any object by using appropriate object detection algorithms and a lenses or web camera. Mobilenetv2 is a neural network architecture that uses depth wise separable convolution as effective building blocks to scan, classify and detect objects from an image or a scene. Mobilenetv2 is used a extractor of classified elements based on their respective feature and provides an efficient mobile oriented model to be used as a base for many image recognition tasks.

Keywords: Computer vision, Artificial intelligence, Object recognition, Neural network, Mobilenetv2

Introduction

The following research paper is constructed taking in view the previous derived techniques related to Object detection. To dive deeper in to the research paper, it is important to better understand the concept of object detection. Object detection is the consolidation of machine learning and deep learning algorithms that detects and define different elements from the image or video received as an input. During the implementation of the object detection model, whenever it detects or scans an object familiar with the one it has been trained with, it draws boxes around those detected elements, making it easier to locate where the objects are and even can track the movement of that element in that particular frame. As mentioned earlier object detection is the combination of machine learning and deep learning, the machine learning aspect deals with characteristic features like color, shape, template etc. and pixels related to an object whereas deep learning aspect do not need the characteristic features but the neural network architecture to perform unsupervised object detection.

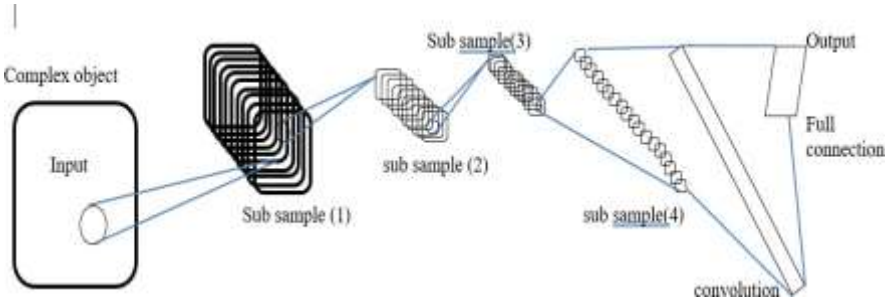

Figure 1. Image segmentation

Upon several tests and comparisons, it was found that models that implemented the neural networks were the most accurate as it can also work in a machine learning aspect and also detected objects at a much faster speed.

The key procedures followed in the object detection process are: -

- When the object detection model is run the web camera of the device starts recording the frame.
- For each input frame we get many complex objects with corresponding classes as output.
- The images captured are taken as an input and is divided into several subsamples.
- Then we consider each sub sample as a separate object.
- Pass on these samples to appropriate algorithms and classify them into various classes.
- As all the samples are classified among different classes, the sub samples are combined to form back the captured image with detected objects.

Object detection techniques:

As discussed, earlier object detection model breaks the input image to several samples and send them through appropriate algorithms to get classified into samples it is important for us to know how the algorithms classify those samples.

- Object detection based on color:

As the name implies it is detection of objects based on their color parameters. In this technique it is important that the color of the object be not an exact match of the color the background. As the input image is breaking into several samples the algorithms compare the samples with different color models. Some of them are CMYK model, LAB model, HSI model, but the most effective of them is RGB model [1].

The RGB model uses the combination of red, green and blue as, these colors can be used to derive almost every other color. The mixture of each color with different intensities and different shade give a total of 16,777,216 possible variable colors.

The CMYK models deploys the use of cyan, magenta, yellow and black colors [14]. This model overlaps the cyan, magenta, a yellow color to create number of other different colors but this combination gives some impurities and to compensate those impurities black color is added.

The LAB color model describes the colors in terms of three parameters. L represents Permanent lightness, and a and b represent four colors: red, green, blue and yellow. The lab color model can be used to locate and communicate with colors.

The samples received the algorithms are compared with one of these color modes and then those are classified according to their set of colors.

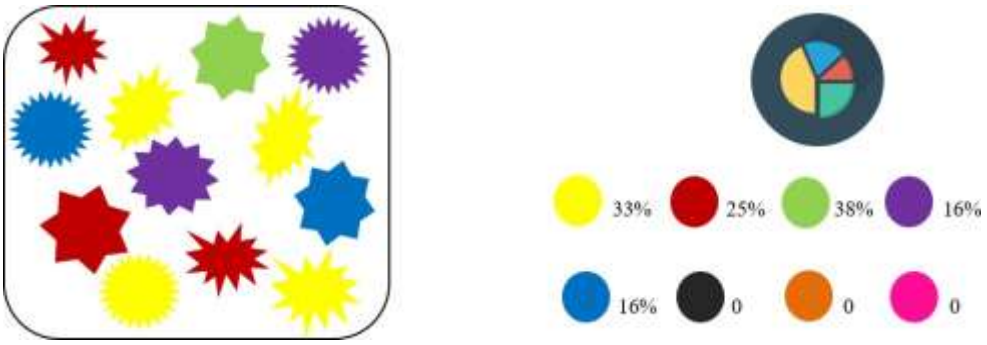

Figure 2. Color based detection

- ❖ Object detection based on shape

As the name suggest, in these objects are scanned for similar shape of object to be detected. It is a highly efficient method as it can be used to detect the object not traceable by human vision [4]

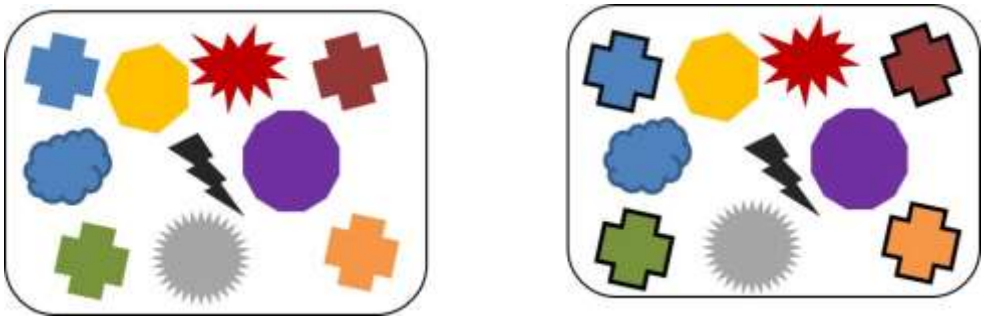

Figure 3. shape based detection

From the above figures, the figure on the left contains different objects with different shapes. And on the right is the same figure but some of the shapes are highlighted with black borders. The image on the left is the input. The algorithm scans the input for the shape to be detected and when the detection process is done the right image is received as an output.

- ❖ Object detection based on template:

This technique is useful for finding small parts of an image which is a match for the template. This process is useful in industries to keep a track on the unit being manufactured. Object detection based on template matching is used to determine the location of (or) trace a particular object in an image or video [2].

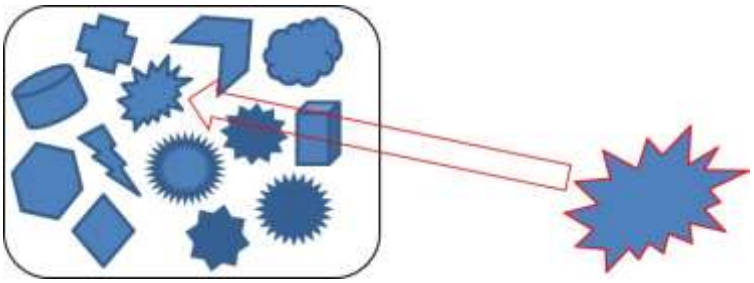

Figure 4. Template Matching

The above figure is an example of an object detection based on template. Suppose the image on the left is received as an input. The image is scanned for a particular element or a template. As the algorithm finds that particular template the object is highlighted or extracted or segmented and is displayed on the output

Earlier approaches:

Object detection can be performed in number of ways depending on the use and type of algorithm. Some of the common detection algorithms include:

- R-CNN
- Fast R-CNN
- Faster R-CNN
- R-FCN
- SSD
- YOLO
- HOG

- **R-CNN**

It is one of the first deep learning algorithm to be bought in use for the object detection system.
R-C NN algorithms involves the use of CNN (Convolutional Neural Network), SVM (Support Vector Machine) and selective search.
CNN is a deep learning algorithm that is used as a feature extractor for the images and acts a classifier for the objects within it. As the CNN algorithm receives input, it extracts each element from it, assign appropriate numeric properties and the classifier analyze those numeric properties and arrange the object accordingly.
SVM is support Vector Machine. The main purpose of SVM is to access whether the element is an object or not and name the class to which the object belongs.
As the input is received, it scans the image and perform selective search on it. Selective search is the process taking an image and scanning it for objects with different sizes, shapes, colors, and textures. This is done by highlighting each object with a bounding box around it [6]. First each element of the image is highlighted with the bounding boxes. Then CNN assigns appropriate numeric values to each element accordingly. The classifier reads the assigned properties and arrange them under categories. The SVM studies the arranged elements. It works on whether the elements belong to any class of object or not. And if it does, it displays the name of class of that element just above the bonding boxes along with percentage of accuracy.

The main drawback with R-CNN algorithm is that on a large image it often overlaps number of bounding boxes on single image due to which the detection process gets extremely slow.

- **Fast R-CNN**

As seen before overlapping of number of boxes over a single image lead to slow detection. To overcome that drawback, Fast R-CNN was introduced. Fast R-CNN algorithm has a region of interest pooling layer and a SoftMax layer on the top of CNN layer.
For each received input feed, there are number of objects with different shapes and sizes, for the R-CNN algorithm, we had to train the network layers separately for different shapes. But the region of pooling technique of Fast R- CNN algorithm converts all such different attributes to a common fixed shape. Due to this the network layers can be trained at a single time [3].
Every element or object of an image has its own set of attributes called feature map and its own region of interest. As the input feed is given to the algorithm, the CNN extracts each element with its attributes called region of interest. The region of interest pooling feature takes each feature map from the input image and the feature map of objects from the data set and create an optical fixed feature map. In this way there would not be any need of separate feature maps.
The SoftMax layer performs the work of CNN as well as classifier.
The SVM is replaced by a linear repressor to give bounding boxes on the detected elements at the output.
The main difference between the R-CNN and Fast R-CNN algorithm is that in R-CNN, the CNN, classifier and the SVM had to be trained separately whereas in Fast R-CNN, the SoftMax layer, and linear repressor are trained at the same time.

The draw back of R-CNN model has also affected the Fast R-CNN model. The selective search used in Fast R-CNN that is ultimately responsible for generation of bounding boxes around the objects also delays the detection of objects due to slow generations of potential bounding boxes. This calls for better technique.

- **Faster R-CNN algorithm**

In faster R-CNN algorithm, there is another new layer added just above the CNN layer known as Region Proposal Network.
In the whole operation of objection detection feature map corresponding toa object is used at two instances, one during the extraction of elements at the first stage (to assign attributes) and second during the final stage to highlight the detected object at the output. In R-CNN and Fast R-CNN, selective search action is performed to generate the feature map at the final stage. And this delayed the whole operation [5].
In the Faster R-CNN, selective search action is not performed, instead of that the feature map generated at the extraction stage is used as the feature maps for the output. And hence the deployed CNN needed to be trained only once.
This feature map is by the addition of fully convolutional network on the CNN layer, this layer performs both action of classifier and CNN and thus resulted with faster object detection.

- **R-FCN algorithm**

Despite of being a whole new layer and less trainable CNN layer, Faster R-CNN model wasn't much faster than the Fast R-CNN, as the algorithm still ran excess of scans during the detection over a single image. To overcome this drawback, R-FCN was introduced.
The R-FCN algorithm uses a position sensitive cropping mechanism. When the image is received as input, the image is cropped into separate regions. Each region has its own feature maps. These feature maps are called position sensitive feature maps [15]. Now in the previous regional based algorithms the feature maps were cropped from the same layer and each cropped section were individually predicted. Whereas in R-FCN algorithms, the feature maps of cropped sections of previous layer were considered. This saved must time and the multiple scanning is not required anymore [7].

- **SSD Algorithms**.

SSD is one-stage detector algorithm useful for the detection of small-scale objects. In SSD algorithm, the bounding boxes generated near the objects are divided numerous different boxes with different aspect ratio. These small boxes show the probability for every feature map location of that image [8]. The SSD algorithm then takes in account all the probability and generates a default box which then detects the objects. Since the probabilities of boxes with different aspect ratio and resolution are taken into account, this algorithm is also known as multireference or multi- resolution algorithm.

- **YOLO**

YOLO is one of the first one-stage, real time object detection technique that uses a single neural network to perform on a single image.
YOLO stands for "You Only Live Once". The algorithms based on the neural networks used classifiers, extractors and repressors to perform detections whereas the YOLO algorithm uses a neural network to divide the image into a number of small fractions [9]. These small fractions are considered as separate images and predicts with bounding boxes around it. YOLO takes into account the whole image during the training and it reasons at a global level when making predictions.
The main advantage of YOLO is that it can perform end to end training and that too with maximum speed of accuracy.

- **HOG detector**

Histogram of oriented gradients is a very important technique used object detection as well as in many self-driving cars. HOG algorithm is a feature descriptor algorithm that removes all the irrelevant data relate to the element of an image like color, position, size etc. and only takes into account the necessary information related to it like shape, edges, gradient etc. required to just identify the element.
In this algorithm, the whole image us broke down into smaller fractions. And for each separate fractions the algorithm calculates the magnitude and direction of edges, gradients, shapes and other relevant attributes [10]. By using the calculated gradient and oriented of these small fractions, the algorithm then creates the histogram.

Proposed methodology:

So far, we have seen the deep learning algorithms used for object detection. Let us look at the methodology on which this research paper is based.

- We have made our model using tensorflow.js and COCO-SSD dataset.
- Tensorflow.js is a JavaScript machine learning library used for making ML models.
- The models made from tensorflow.js library can be accessed via web browser as well as node.js
- We have used the Mobilenetv2 as the feature extractor along with COCO-SSD dataset.
- COCO is a world-wide object detection dataset that has nearly 80 different object categories, 91 stuff categories, 250,000 people with key points, 1.5 million object instances, and nearly 330 thousand images.
- The model is based on the training of mobile net neural network by using the single shot detectors.
- SSD is one-stage detector algorithm useful for the detection of small-scale objects. In SSD algorithm, the bounding boxes generated near the objects are divided numerous different boxes with different aspect ratio [11]. These small boxes show the probability for every feature map location of that image. The SSD algorithm then takes in account all the probability and generates a default box which then detects the objects. Since the probabilities of boxes with different aspect ratio and resolution are taken into account, this algorithm is also

known as multireference or multi-resolution algorithm

- As we run the JavaScript code, the input feed received by the camera is scanned by the mobilenetv2 framework [13]. The image is divided into several different pieces with their own feature map. Several bounding boxes are generated around the small pieces. The feature extractor used, extracts the small pieces and compares with the COCO dataset [12]. The bounding boxes associated with the small pieces start showing predictive probabilities. The SSD algorithm then takes in account all the probability and generatesa default box which then detects the objects.

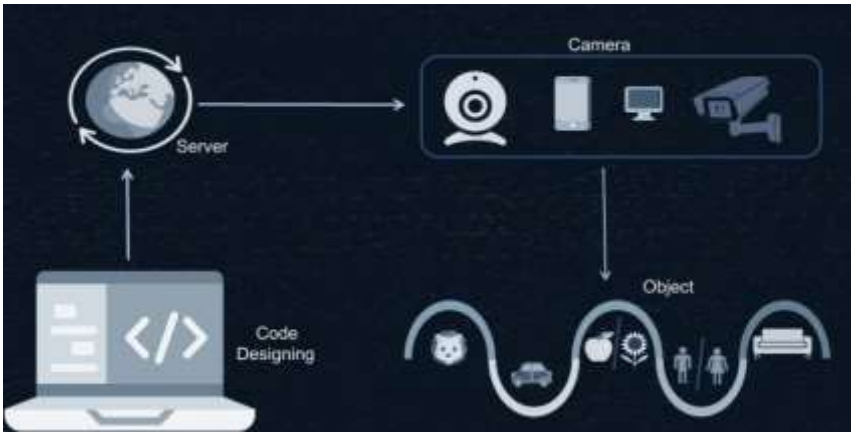

Figure 5. Block diagram of proposed Object Detection Model

Benefits after its implementation:

- **Optical character recognition**

Optical character recognition is used for extracting data from a printed or written format or image and translating the text into machine language for further data processing.
The OCR works like an object detection system. When the camera of OCR scans or reads a text from printed, written or even jpeg form, it scans for each and every alphabet and it displays whatever is written there and directly convert the alphabets to machine readable language

Figure 6. OCR using Object Detection

- **Self-driving cars**

Self-driving cars the modern innovation that doesn't require any human intervention for driving it drives completely by itself using artificial intelligence and machine learning. The self-driving cars are capable of sensing the surrounding

areas and as soon as they sensors pickup signals of any obstacle on its way, the car will work accordingly. Self-driving cars uses the object detection techniques to continuously keep track of the surrounding areas, it

Figure 7. Self Driving Cars

- **Tracking objects**

Tracking objects with an object detection framework is also possible for example, tracking a ball during a cricket match, tracking a person in a video, tracking the ball in a football match etc. The main example of tracking using object detection is tracking a car using surveillance cameras. When required the surveillance cameras can also be programmed to track a particular car in heavy traffic and it will be keeping a tag of locations at which the car was spotted. It is needless to say that object detection will be even more critical in the field of protection and surveillance.

The object detection can also be used to find a particular object from a set of any.

Figure 8. Tracking of Objects

- **Face detection and face recognition**

Almost everything around employs the use of object detection. The smartphone that we use on a daily basis has a feature of face recognition that is nothing but an application of object detection system. When we are setting the face recognition in our smartphone, we give various frames of images of face. Those images are sed as a dataset and every time we show our face to smartphone camera the object detection system runs the received input feed against images of our face stored in a dataset.

The filters that are now a days available in many social media apps also use object detection and recognition.

Figure 9. Dace detection using Object detection

- **Activity recognition**

Object detection can also be used to study the movements of life forms. This is generally used in surveillance cameras, cameras installed at jails and many areas. The object detection system scans video its is receiving as a input and it scans the activity of different moving elements. The major example of activity recognition is in the field of bio science. The object detection system can be used to track the activity of seed growing, track the behavior of animals on which different vaccines are tested and also to train the robots.

Figure 10. Activity Recognition using Object detection

- **Object extraction from an image of a video feed**

The simple example of object extraction is the scanning of bar codes at shops, object extraction refers to the area segmentation of desired part of an image or a video. The object extraction does not mean that the object will be takenout of that image. This only highlight and magnifies the object required.

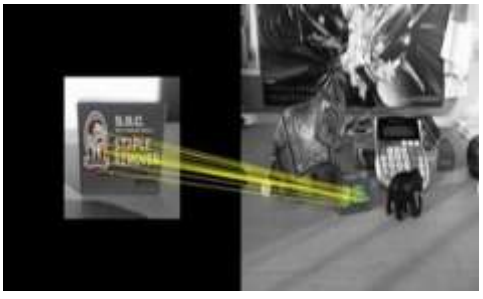

Figure 11. Object extraction

Future scopes:

The science of Object detection won't be fading anytime soon, almost 70% of the daily advancement is in the field of computer vision and artificial intelligence only. Let us look at future scopes of object detection system:

- **Military / defense system**.

Object detection can be used to track the movement at the enemy lines, in future bots can be made to fight the wars that we as a human cant in that case object detection will be of great asset regarding tracking and observing the enemy lines.

- **Robotics**

The days are not far when robots will be interacting with humans almost at every instant of a day. In that case object detection will help the robots to study the human activity and work more efficiently.

- **Developing Route Maps**

As per recent studies, 78% earth is still unknown to humans that is we still don't know the locations and directions of most of the earth terrains. Object detection van be used by satellites to construct the route maps of much possible terrains and also update the pre-existing maps.
Object detection can also be used in whether forecasting and can be used to get prior warnings before any climate change can actually happen.

- **Observe movements of outer space**

Object detection can also be used to study the changes in outer space that includes movement of stars, asteroids, planets, comets etc. in this way we can train a model to get prior warnings before any space event.

- **Education**

Object detection can be used to enhance the teaching procedure, deriving a much better, engaging and interactive teaching mechanism. Object detection can be used develop a teaching platform for dumb and deaf people. A system that can understand the sign language of dumb and deaf people and in return communicate back

Result and conclusion:

At the end of this research paper we have covered the basic concepts and techniques of object detection the object detection algorithm namely YOLO, R-CNN, R-FCN etc. have also been covered. A JavaScript model with tensor flow library was introduced that used mobilenetv2 as feature extractor and SSD algorithms were used. For the reference of the object COCO dataset was used. At the end the model showed promising results with much better accuracy and fast detection. While the model, we used 24 fps of frame rate.

References:

1. Paul Viola Michael J. Jones, "Robust Real-time Object Detection", Technical Report Series Cambridge Research Laboratory, (February 2001), CRL 2001/01

2. Papageorgiou, C., Poggio, T. A Trainable System for Object Detection. International Journal of Computer Vision 38, 15–33 (2000)

3. Peng Zhou, Bingbing Ni, Cong Geng, Jianguo Hu, Yi Xu; “Scale-Transferrable Object Detection”, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 528- 537

4. S. K. Divvala, D. Hoiem, J. H. Hays, A. A. Efros and M. Hebert, "An empirical study of context in object detection," 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1271-1278, doi: 10.1109/CVPR.2009.5206532

5. P. F. Felzenszwalb, R. B. Girshick, D. McAllester and D. Ramanan, "Object Detection withDiscriminatively Trained Part-Based Models," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 9, pp. 1627-1645, Sept. 2010, doi: 10.1109/TPAMI.2009.167.

6. Jiahui Yu, Yuning Jiang, Zhangyang Wang, Zhimin Cao, and Thomas Huang. 2016. UnitBox: An Advanced Object Detection Network. In Proceedings of the 24th ACM international conference on Multimedia (MM '16). Association for Computing Machinery, New York, NY, USA, 516–520

7. C. Vondrick, A. Khosla, T. Malisiewicz, A. Torralba. "HOGgles: Visualizing Object Detection Features" International Conference on Computer Vision (ICCV), Sydney, Australia, December 2013.

8. Kong, T., Yao, A., Chen, Y., & Sun, F. (2016). HyperNet: Towards Accurate Region Proposal Generation and Joint Object Detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 845-853

9. Edgar Osuna, Robert Freund, and Federico Girosi. Training support vector machines: an application to face detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1997.

10. C. Papageorgiou, M. Oren, and T. Poggio. A general framework for object detection. In International Conference on Computer Vision, 1998.

11. J. Quinlan. Induction of decision trees. Machine Learning, 1:81–106, 1986.

12. D. Roth, M. Yang, and N. Ahuja. A snowbased face detector. In Neural Information Processing 12, 2000.

13. H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. In IEEE Patt. Anal. Mach. Intell., volume 20, pages 22–38, 1998

14. R. E. Schapire, Y. Freund, P. Bartlett, and W. S. Lee. Boosting the margin: a new explanation for the effectiveness of voting methods. Ann. Stat., 26(5):1651–1686, 1998.

15. Robert E. Schapire, Yoav Freund, Peter Bartlett, and Wee Sun Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. In Proceedings of the Fourteenth International Conference on Machine Learning, 1997