Available online at www.sciencedirect.com

SCIENCE DIRECT®

Pattern Recognition Letters 27 (2006) 1275–1288

Pattern Recognition Letters

www.elsevier.com/locate/patrec

Pragmatic Genetic Programming strategy for the problem of vehicle detection in airborne reconnaissance

Daniel Howard [a,*], Simon C. Roberts [a], Conor Ryan [b]

[a] QinetiQ, Malvern Technology Centre, St. Andrews Road, Malvern, Worcestershire WR14 3PS, United Kingdom
[b] Computer Science and Information Systems Department, University of Limerick, Limerick, Ireland

Available online 28 December 2005

Abstract

A Genetic Programming (GP) method uses multiple runs, data decomposition stages, to evolve a hierarchical set of vehicle detectors for the automated inspection of infrared line scan imagery that has been obtained by a low flying aircraft. The performance on the scheme using two different sets of GP terminals (all are rotationally invariant statistics of pixel data) is compared on 10 images. The discrete Fourier transform set is found to be marginally superior to the simpler statistics set that includes an edge detector. An analysis of detector formulae provides insight on vehicle detection principles. In addition, a promising family of algorithms that take advantage of the GP method's ability to prescribe an advantageous solution architecture is developed as a post-processor. These algorithms selectively reduce false alarms by exploring context, and determine the amount of contextual information that is required for this task.
© 2005 Published by Elsevier B.V.

Keywords: Genetic Programming; Object detection; Method of stages; Reconnaissance; Discrete Fourier transform; Vehicle detection; Machine vision

1. Motivation for this research

Our aim is to produce software that scrutinizes a digital image to detect a class of object. This automation aims to reduce the human effort of visually scouring through large amounts of imagery. It does not aim to entirely substitute the scrutiny of imagery by human photographic interpreters (PIs) but to selectively reduce the volume of imagery that merits their analysis. The computer assisted cuing of PIs exploits more imagery in the allotted time and alleviates PI oversight and fatigue. It may help to reduce visual illusion errors by PIs because such scrutiny algorithms are often defeated (fooled) in different ways to our own human vision.

Targets of this visual search cannot be precisely described and can only be described imprecisely by a linguistic label (Zadeh, 1999). Therefore, the research question is how best to automate the detection of a class of target of high variability and of imprecise definition? Our contribution to this broad research objective tackled the problem of detection of an imprecise class of target (evidence indicating the presence of any vehicle) in a 'line search' of infrared imagery (infrared line scan, IRLS). This 'line search' was a long run of continuous imagery taken over parts of southern England by a low flying aircraft. The data set for this problem is described in Section 2.

In this representative problem, a 'vehicle' can be imprecisely defined as an assembly of sufficient 'vehicle clues'. It is vital to understand that this is not a 'toy' problem with a right or wrong answer, and algorithms are judged by the value that they bring to the PI users, i.e., one cannot meaningfully measure the performance of the algorithm by counting false positives and false negatives as these are imprecisely defined and often subjective (Fig. 5). Moreover, the application of interest was not to determine the number of objects (vehicles) in an image for which other techniques are more suitable. In this surveillance application, PIs are interested in all manner of vehicle clues because vehicles

* Corresponding author.
E-mail address: dhoward@taz.qinetiq.com (D. Howard).

0167-8655/$ - see front matter © 2005 Published by Elsevier B.V.
doi:10.1016/j.patrec.2005.07.025

may be disguised. The algorithms may detect vehicles that have recently left the scene (thermal signatures), and partially hidden vehicles. The value of the automation is not only measured by time saved but by the chance discovery of something that might make the whole scrutiny effort worth while.

2. Imagery data

A 25 mile run over the south of England produced a 'line search' of IRLS infrared imagery that contained all manner of environments (urban, industrial, rural) and some environmental variability (patches of rain). Vehicles appear almost anywhere: drives, roads, off-road, gateways, parking areas, hidden by the canopy of trees, partially obscured or in the shadow of neighboring buildings and thermal shadows of departed vehicles. Vehicles can appear in a number of states: hot and cold engines. The size and length of vehicles can vary: viewed from above but the roll of the aircraft and camera width permits moderate angles of perspective.

The imagery was retained in its raw format, consisting of 8 bit grey scale pixels and containing repeated lines from the aspect ratio correction in IRLS line scan. Ephemeris data (information about the altitude and roll of the aircraft) was available for each IRLS line and when missing, the data from the previous line was assumed. This single continuous image (the line search) was divided into 27 images each with a width of 3072 pixels and an average length of 7066 pixels or 22 million pixels per image. Aircraft altitudes ranged between 304 ft and 668 ft. Images have ID labels and Table 1 describes the properties of these images. Fig. 1 gives various examples of thermal signatures for vehicles in this imagery.

Copious visual inspection effort identified the regions containing vehicle clues to produce a 'truth' for this large amount of data. Vehicles were subjectively sorted into categories: clear, subtle, faint. A special class denoted portions of the imagery that contained objects that strongly resembled vehicles (e.g., ships in canals). It was left out of the 'truth' and out of the inductive learning (with the rationale that the user is ambivalent about detection of these objects). Roughly 600 vehicles in these images were 'marked up' by this procedure and imagery was then divided into a training set, a validation set, and a test set of images.

3. A Staged Genetic Programming method

Genetic Programming (GP) runs evolve vehicle detectors by using examples of 'object' and 'non-object'. Detectors are algebraic functions (and logical functions) of pixel information and are applied pixel by pixel (although our production demonstrator uses pixel jumps) to detect vehicles in new imagery. A multi-stage GP method was first presented in (Howard et al., 1999) to evolve fast and accurate detectors in short evolution times. Given an image set of size $N \times M = O(10^7)$ pixels containing objects and their immediate surroundings that can be bounded in sub-windows of size $n \times n = O(10^2)$ pixels, the task is to construct a function of pixel data with support $n \times n$. An object is detected at a pixel when the function evaluation is positive. A number of images is selected with a 'truth' of known object locations prepared for each image.

Evolution of the detector to discriminate the object pixels from all of the non-object pixels in the image would involve evaluations at all of the $N \times M$ pixels and would be prohibitively expensive. Instead, a first stage of GP takes a random selection of non-object pixels and all the object pixels from the truth as test points. The fittest detector from this evolution stage is applied to the $N \times M$ pixels of each image producing a set of false positives (FP). A second stage of GP uses the discovered FP and all of the object pixels from the 'truth' as test points to evolve a second detector. It has a tough job because it must discriminate like from like. The fittest detectors out of both GP stages are combined, i.e., an object is detected only when both return a positive value. The first detector is applied

Table 1
Image information for the image set, R stands for a right aircraft roll and L for a left aircraft roll

ID	Altitude (ft)		Roll (°)		x size (pixels)	y size (pixels)	ID	Altitude (ft)		Roll (°)		x size (pixels)	y size (pixels)
	Min	Max	Min	Max				Min	Max	Min	Max		
2	392	448	0	48.9R	3072	7772	24	480	544	0	0	3072	6582
4	408	436	0	12.3R	3072	8007	28	240	260	0	0	3072	13,844
5	320	372	0.6L	0	3072	9821	47	496	556	0	7.6R	3072	5974
7	304	340	0	2.8R	3072	10,954	48	424	444	0	4.7R	3072	5451
8	296	320	0	0	3072	11,305	49	440	444	0	6.7R	3072	5450
10	304	308	0	0	3072	11,259	50	408	436	0	10.3R	3072	5800
13	440	528	0	44.2R	3072	5950	51	404	408	2.5L	0	3072	8616
14	372	440	0	0	3072	6836	52	580	588	0	0	3072	5971
15	376	380	0	0	3072	6698	53	560	580	0.5L	0	3072	5590
16	372	376	0	0	3072	5410	54	540	556	0	0.6R	3072	5755
19	392	488	0	0	3072	6573	58	548	564	0	0	3072	6163
20	500	500	16.8L	0	3072	3645	59	608	660	0	0	3072	4971
21	500	508	6.1L	0	3072	6490	60	660	668	0	2.7R	3072	5157
23	452	476	0	0	3072	4729							

Download English Version:

https://daneshyari.com/en/article/535742

Download Persian Version:

https://daneshyari.com/article/535742

Daneshyari.com