AN ARTIFICIAL INTELLIGENCE APPLICATION FOR ARCHITECTURAL TEXTURES ANALYSIS

D. G. Papi

Dept. CSA, Milan Institute of Technology, via Bonardi 3, 20133 Milan Italy, daniele.papi@polimi.it

KEY WORDS: Artificial Intelligence, Architecture, Texture Analysis.

ABSTRACT:

The application of AI engines into the automatic (unsupervised) or semiautomatic (supervised) analysis of the architectural textures is as obvious as unpredictable. The application we present in this paper is a custom-made engine which has been developed basing upon a popular and well known application pack. The architectural textures present various peculiar elements which require a different approach than usual to most sides of the matter. We use the complex feature definition (vectors and raster surfaces are seen as components of the same object) extending its meaning to the architectural texture itself.
Under the definition of “architectural texture” we group the result of the processing ofa very wide group of multiscale images concerning the whole family of the “designed matter”. This means that we consider everything, from land scale interventions to house making, as the object of our research. The connecting wire among seemingly such distant items should be just architecture texture itself, in order to use it as the unique valid measuring and comparison element for designing, preserving and classifying the architectural matter.
The creation by a artificial intelligence group of routine of the architectural texture database is made for futher AI engine feeding purposes, for making a try to structure a way to automatic architecture analysis. Next step of this research plot will be unsupervised architecture classification for design support and for city-planning and building lawmaking.

1.THE APPLICATION

1.1 Introduction

The borders between an AI application and any other advanced automatic function are normally defined by its ability to acquire information and improve its efficiency by processing data. Actually, we normally speak of "training phase" at the start-up of any AI project, applying a kind of human behaviour to the machine. This is just a word misuse, obviously, but in spite of this evidence, in the history of automation many have used electronic digital equipment to perform some typically human activity with so good results that the definition "Artificial Intelligence" got a real kind of sense.
In our application we used a programmed computer for trying to obtain quite a "critical" performance: it consists in a characteristic human intelligence application that normally requires some analytical and associative intelligence, such as analysing manufactured textures within architectural objects. We think this can be considered a good testing field because "texture" is an important architectural element by itself and is not necessarily a recursive, repetitive pattern coating a pure geometrical structure without interacting with it. Many architecture critics have written about the relevance of texture among other defining terms of projected and built matter and the contemporary specific literature keeps texture in the roster of the main topics for defining and analysing architecture.

1.2 Definitions

The problem of analysing spatially and time dependent data occurs in many different fields: economics, sociology, ecology and environment management, agriculture, hydrology, engineering and finally architecture. As a matter of fact time (counted in years) is not always a primary aspect of architectural objects analysis, outside of historical sedimentation reconstruction within restoration projects, but in our application we can consider time (counted in seconds) as a support data set, which will turn useful to interpret ambiguous read-outs from dynamic imagery sets. Ambiguity, in our case, is very frequent, so we will consider texture analysis as a fuzzy problems and its data will be grouped into non-Cantor sets. The AI application will consider geometrical data as a fuzzy sets and the results of their cross evaluation will give architectural pattern evaluation. The result of the evaluation is a catalogue of architectural objects grouped into different overlapping sets. The final goal of this research is to process architectural patterns data to perform an important phase of a wider morphological analysis of architectural objets, aiming to organise a model for project activity and planning, also from a normative point of view. Obviously, the different nature of the architectural objects in the training phase (catalogue building), will give different results: this can be seen as the correct consequence of the different local habits, tastes and uses and will generate different (not necessarily slightly but locally fully valid) project and lawmaking support.
So, what is exactly the "raw material" we want to process? The word texture (The Collins English Dictionary, 2000, HarperCollins Publishers) is defined as follow:

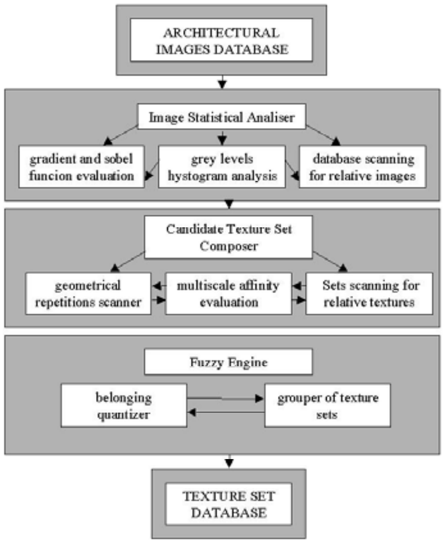
1 *the surface of a material, esp. as perceived by the sense of touch example: a wall with a rough texture*
2 *the structure, appearance, and feel of a woven fabric*
3 *the general structure and disposition of the constituent parts of something example: the texture of a cake*
4 *the distinctive character or quality of something example: the texture of life in America*
5 *the nature of a surface other than smooth example: woollen cloth has plenty of texture*
6 *(Art) the representation of the nature of a surface example: the painter caught the grainy texture of the sand*
7 *(Music) a. music considered as the interrelationship between the horizontally presented aspects of melody and rhythm and the vertically represented aspect of*

harmony, example: a contrapuntal texture - b. the nature and quality of the instrumentation of a passage, piece, etc.

All these definitions have something to do with architecture and are useful to give account of the perimeter of our application aims.
When talking of architectural texture we consider that architecture is not a natural environment and that it depends on the expression of a projectual will. This strongly (and positively) influences the freedom that elements have to be considered as "architectural texture".
The point that really need to be stressed is the great prominence that metric scale has in this evaluation. David Chipperfield talks of "architectural texture" meaning that "texture" is the result of all those projectual actions that cohoperate to build a "skin" to architectural objects. Leonardo Benevolo talks of "architectural texture" giving this expression the sense of "artificial modifications that the presence of architectural objects causes on the surface of the land", defining it just as "the signs of man on the Earth".
So, architectural texture and non-naturality are, in any case, in a very close relationship. This relationship is just what gives its aspect to the human activity world at any scale, from room to town.
This textures have a general connection: they belong to the project sphere and because of this they are influenced by styles and ages, political conditions, psychological aspects, society rules and have a kind of unintentional dependence from their framework.
This involuntary common aspect are those we look for, because they need to be scanned, known and used to support future projectual actions, either from a purely factual point of view and from a legislative one.
This last has a special importance because it would allow to give the designers a set of rules that derive from their own field analysis, using architecture itself as the ruler of architecture. This is impossible to obtain with the usual building law, which are written only for function, structure and zoning.
If we want to use a programmed PC to give project and law-making support by organising a critical catalogue of architectural objects and we also want to obtain this result basing onto the different aspects of "architectural texture" (with a special regard for its purely geometrical aspects [Alvarez and Morales1997]), the problem now must be split in two different branches.
First: what will be the abacus of this "catalogue"?
Second: how and which way architectural textures can be analysed by a PC?

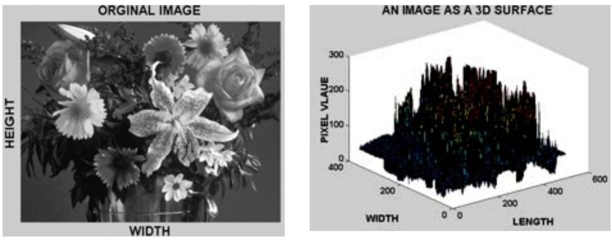
1.3 The catalogue

The architectural textures we consider are presented as 8bit greyscale bitmaps of a standard size of 2048x2048 pixels. This size is constant, regardless the real dimensions of the elements. Therefore, this results in a virtually homogeneous set of images that contain either the "skin and the shape" of the analysed architecture.
The complexity of the whole procedure increases in direct proportion with the number of images in the set, mainly as far as the "coding" problems are concerned. "Coding finder" is the name of a routine that uses as a principle to group images their affinity in geometrical terms.
The shape of planimetric views of European urban fringes, for instance, have often shown to have a quite strong correspondence with architectural geometry of the façades of the buildings that constitute the same areas.
This has revealed not to be true if the test was performed in north American Midwest zones. A geometric affinity "pole" called "Euroshape" has been included into the sets for a possible insertion in the catalogue.
The process is unsupervised - except for the naming procedure. This catalogue of texture reference sets is built up by a standard AI process called "stratified inference" [Ballard1991] [Barnard1983]. The scheme of the flow chart of our version of this very flexible algorithm is the following:



When a texture element is output as a possible catalogue pole, a "texture set" is created, it receives a name and becomes a "qualified set". Hundreds of such sets are created and stored for further processing.
When all the possible elements have been evaluated, what we call the "protocatalogue" is ready to be processed by a modified "fuzzy spreader" algorithm. It performs the job of building an unsupervised fuzzy set grouping of all the "similar" elements in the protocatalogue, generating a "belonging intensity hystogram" for each of them. The aim is to define the
Being fuzzy logic an extension of conventional boolean logic, it can handle partial truth such as "low", "med-low", "med", "med-high", "high" and all of the intermediate levels. In this application we quantize the "appartenence intensity on a scale of 256 different levels. This limit derives from the evaluation of the screening ability of an average human vision system - about 200 different light reflection intensities can be perceived. This is particularly important and it's not a simply "standard choice" because architecture itself "talks" in the visible range. The coincidence with the 256 different grey levels expressed by the standard 8bit file format is actually a lucky condition but nothing more.
We know that from the very beginning, when in 1965 Lotfi Zadeh published the paper "Fuzzy Sets" in the journal of Information and Control, fuzzy logic is described as empirically based. It relies on the user's experience rather than the technical understanding of a problem and this makes fuzzy rules extremely fitting for AI applications.
To build ours, we used an evaluation version of a very popular software. The Cobalt A.I. SDK Principle Component Analysis (PCA) system uses just fuzzy logic to determine relationships between data.
This fuzzy engine is used to build the belonging intensity histogram of each element.

The method proposed here is based on fractal dimension and is derived from Soundararajan Ezekiel and John A Cross work. They hypothesize the fractal dimension correlates with image roughness and is very useful to consider the geometrical recursivity of artificial textures, such as architectural ones.
Fractal-based texture analysis is not a new deal as it was introduced by Pentland in 1984. Soundarajan and Cross defined the method as follow.



Let T, FD and H be the topological dimension, fractal dimension and the Hurst exponent [7][9]. For images, T=3 because there are two spatial dimensions and the third dimension is image density. Figure 1 shows the original image and its 3-dimensional representation.
The parameters H and FD can be estimated from

$$E[\Delta^2 f] = c[\Delta^H d]^2$$

where E, Δf, Δd , and c are the expectation operator, intensity operation, spatial distance, and scaling constant.
Substitute H=3-FD, and $\kappa = E(|\Delta f|)_{\Delta d=1}$ in the above equation, we have

$$E(|\Delta f|) = \kappa \Delta d^H$$

By applying log to both sides we have

$$\log E(|\Delta f|) = \log \kappa + H \log(\Delta d).$$

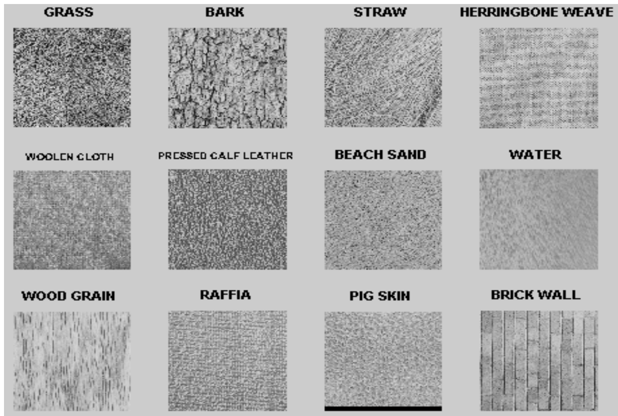
The Hurst exponent H can be obtained by using the least-squares linear regression to estimate the slope of the grey-level difference *GD (k)* versus k in log-log scales and k varies from 1 to the maximum value s where

$$GD(k) = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N-k-1}|I(i,j) - I(i,j+k)| + \sum_{i=1}^{N-k-1}\sum_{j=1}^{N}|I(i,j) - I(i+k,j)|}{2N(N-k-1)}$$

The fractal dimension FD can be derived from the relation *FD=3-H*.
The approximation error of the regression line fit should be determined to prove that the analysed texture is fractal, and thus be efficiently described using fractal measures. A small value of the fractal dimension FD, implies to a the large value of the Hurst exponent H represents fine texture, while a large FD, implies to a smaller H value, corresponds to the coarse texture.
The imagery set of training elements has been structured using the base concept of "complex feature" [see Papi,2002] where the fractal elements can be considered as showing either the geometric recusivity parameter or the coarse index of the texture. In architectural textures the complexity is not casual and the geometry is strictly connected to appearing roughness. The evaluation passes to the fuzzy engine which starts with assembling the belonging intensity histograms.
The different intensity histograms have been limited to just 3 dimensions, but this limit is only due to calculation limits and has nothing to do with real space or with any peculiar constrain.
Architectural textures are actually defined under those three dimension: geometrical relationships among inner parts; internal recursivity of components, radiometry proportions (grayscale).

Brodatz images test

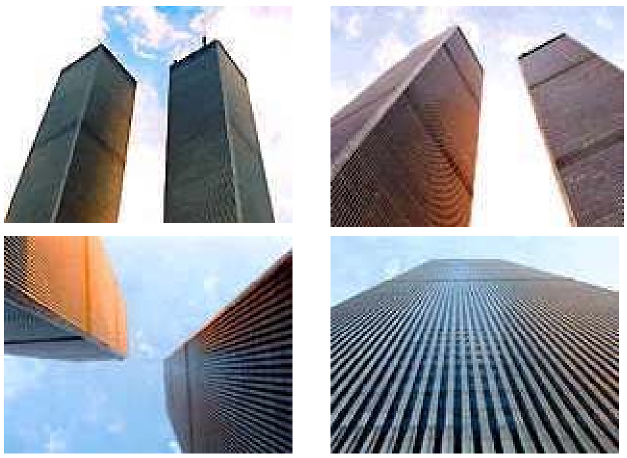
The original fractal classification with the Soundarajan and Cross method gives the following results. The Hurst coefficient and the fractal dimension are evaluated too.
Standard Brodatz images of size 512 by 512 were used for classification test. They were Grass, Bark, Straw, Herringbone weave, Woolen cloth, Pressed calf leather, Beach sand, Wood grain, Raffia, Pigskin, and Brick wall.



These are the results obtained with the Soundarajan and Cross fractal method:

Grass	Bark	Straw	Weave	Cloth	Leather
2. 6571	2. 5494	2. 6881	2. 6323	2. 7170	2.6884
B.Sand	Water	Woodgrn	Raffia	Pigskin	Brickwall
2. 6432	2. 6827	2. 7256	2. 5665	2. 6142	2. 7039

The architectonic textures we have processed give significantly improved results as their recursivity we assumed as an a-priori condition showed to be a fact. The following images has been grouped by the fuzzy engine after the evaluation of their internal level of recursivity and fractal coefficient.



This is the obtained group named “towers”. The fuzzy engine also missed the following possible hits, because of their geometric difference, that is not a “conceptual difference” and can be considered ad a mistaken choice basing on the category choice we tried to describe as a constrain.

Missed hits

Another very small group called “bridge” was obtained, but the hit miss rate in the pre-selected images was so poor that we only show the grouped elements

Conclusions

The artificial intelligence algorithms implemented using Cobalt SDK code are functional to our aim as far as a smart pre-supervision on the candidates is made. Natural intelligence can perform so “elemental and basic” actions that the artificial one is cut out of having any chance to manage the same.
For instance, no routine can say if a raster file containing 2048x2048 pixels contains also an urban sector where people lives easily, while a city planner eye just needs a blink.
This kind of limit is still not surpassed by standard code assemblers and languages. Probably this topic still needs a large amount of research and efforts from architects and engineers involved in town and architecture analysis, but we think that most part of the job should be probably done by computer scientists. Software engineers, mathematics and data processing experts are still on the way for developing, testing, producing and finally giving the community those really performing “standard instruments” that are still missing in this peculiar application.

References

L. Alvarez y F. Morales, 1997, Affine morphological multiscale analysis of corners and multiple junctions. International Journal of Computer Vision, 25(2):95--107, 1997.
Asada and Brady, 1986, The curvature primal sketch. IEEE Trans. on Pattern Analysis and Machine Intelligence, 8(1):2—14.
S. Barnard, 1983, Interpreting perspective images. Artificial Intelligence, 21,.
K. Brunstrom, J. Eklundh, y T. Uhlin,1996, Active fixation for scene exploration. International Journal of Computer Vision, 17(2), February 1996.
J. Canny, 1986. A computational approach to edge detection. IEEE Trans. on Pattern Analysis and Machine Intelligence, 8(6):679--698, November 1986.
D. Geman and B. Jedynak, 1986. An active testing model for tracking roads in satellite images. IEEE tran. on Pattern Analysis and Machine Intelligence, 18(1), January 1996.
G. Giraudon and R. Deriche, 1991. On corner and vertex detection. In Proc of the Conference on Computer Vision and Pattern Recognition, pages 650—655
T. Lindeberg and M. Li., 1997, Segmentation and clssification of edges using minimum description length approximation and complementary junction cues. Computer Vision and Image Understanding, 67(1):88--98, July.
D.G. Papi, 2002, Representation and Analysis of Archaeological Elements by an Original Digital Method, DPI, Milan, 2002
L. Parida, D. Geiger, and R. Hummel. Kona, 1997, A multi-junction detector using minimum description length principle. In Proc of the Energy Minimisation Methods in Computer Vision and Pattern Recognition Workshop, pages 51--65.
L. Parida, D. Geiger, and R. Hummel, 1998, Junctions: Detection, classification, and reconstruction. IEEE Trans. on Pattern Analysis and Machine Intelligence, 20(7).
A. Rattarangsi and R.T. Chin, 1992, Scale-based detection of corners of planar curves. IEEE Trans. on Pattern Analysis and Machine Intelligence, 14(4):430--449.
S. Sarkar and K.L. Boyer, 1993, Perceptual organisation in computer vision: a review and a proposal for a classificatory structure. IEEE Trans. on Systems, Man, and Cybernetics, 23:382--399.
J.A. Shufelt, 1999, Performance evaluation and analysis of monocular building extraction from aerial imagery. IEEE Trans. on Pattern Analysis and Machine Intelligence, 21.
J.A. Shufelt, 1999 Performance evaluation and analysis of vanishing point detection techniques. IEEE Trans. on Pattern Analysis and Machine Intelligence, 21.
J. Zerubia and R. Chellapa, 1993, Mean field approximation using compound gauss Markov random fields for edge detection and image estimation. IEEE Trans. on Neural Networks, 8(4):703--709.
S.C. Zhu, Y. Wu, and D. Mumford, 1997, Minimax entropy principle and its applications to texture modeling. Neural Computation, 9(8).