Evomorph: morphological modularization in A.I. for machine vision inspired by embryology

Daniel Howard
Howard Science Ltd
Malvern, UK
dr.daniel.howard@gmail.com

***Abstract*— Nature likely implements modularization in multicellular developmental biology using the chemical context of the cell, cell division generational distance, and genetic triggers. Inspired in this, Evomorph is a proposed heuristic method of Artificial Intelligence that pairs these concepts with Evolutionary Computation. It is described here as a flexible template matching for object detection in Machine Vision.**

Keywords*— *embryology, modularization, machine vision, image analysis, object detection, classification, template matching, pattern matching, Artificial Intelligence, Genetic Programming, Evolutionary Computation, code re-use.

I. INTRODUCTION

An apparent code reuse is observed in many repeated patterns in Nature. In particular, one sees repetition of shape in a multicellular life form such as our eyes, ears and limbs.

Biological research has confirmed that Turing's diffusion driven instability [1] specializes this modularity resulting in slight differences between modular shapes. Complex patterns can be established by slow activation and faster inhibition altering the re-used component. Although these ideas are not yet comprehensively adopted, in future, they have scope to greatly enhance the method that is proposed in this paper.

II. INSPIRED IN EMBRYOLOGY

Figure 1 illustrates a simplistic model of developmental biology that inspires the method. It consists of three elements: (1) cell DNA, (2) communicated chemistry and (3) external stimuli. It encompasses basic ideas from [2].

A single cell produces a daughter cell, and the daughter cell then produces a grandchild cell and so on. Defining cell division generational distance as a count of the divisions that took place from the initial cell, the chemical context of each cell is a function of the chemical context of the parent cell and becomes altered with distance. Both intracellular chemicals and external stimuli will determine the activated instruction of the DNA, which gene is invoked or fired.

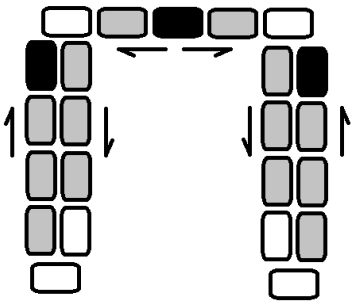

Figure 1. Cell division that achieves modularization

The changing chemical context is inspired in the Biology of Gap Junctions, intercellular connections that directly bridge the cytoplasm of two cells. These allow various molecules, ions and electrical impulses to directly pass through a regulated gate here between adjoining cells.

The dark cells in Figure 1 represent one start and two stop cells. The initiator cell is in the centre of this diagram. This cell is special in that it produces two or more daughter cells. A multicellular structure emerges by cell division. To simplify the presentation, assume that the external stimuli is absent. As the organism grows, the chemical context of distant relatives becomes different from that of the progenitor cell. Under no external stimuli, and if maintaining symmetry then at some generational distance a different gene is triggered to alter the direction of growth. In this diagram cells that change the direction of growth are shown as clear (not shaded).

Two symmetrical structures, equivalent to limbs in a real multicellular organism emerge from this process. Note that this is unlike methods of Artificial Intelligence such as Genetic Programming (GP) that evolve the implicit or explicit modularizations and their reuse [5-12]. In the proposed model the modularization concerns shape and morphology of the organism. In Figure 1, it is cell division distance dictated by the chemical context of the cell that determines when and where to invoke reuse.

This computational model is but a gross simplification of the true mechanics of cell differentiation from stem cells into non-stem cells and related symmetric and asymmetric cell division. In Figure 1, the clear (not shaded) cells indicate a point of differentiation between cells. It is the point at which sufficiently altered DNA expression represents a significant change, a change that exceeds a threshold, a trigger. An enriched version of this model would permanently change DNA expression in the daughter cells. The algorithm is inspired in the mechanics of cell biology and embryonic development as described by (Trosko, personal communication, [2]):

"Pursuant to exogenous factors, the normal organ specific stem cell can divide symmetrically or asymmetrically, with cancer cells having lost this ability to respond to those exogenous factors, dividing only symmetrically. If the body needs more stem cells, then the stem cell divides symmetrically. If the body needs more differentiated cells, then the stem cell divides asymmetrically upon receiving a critical signal. The "initiated stem cell", the normal cell that is altered when a critical gene is mutated as a response to the asymmetrical division inducing exogenous signal can, from then on, only divide symmetrically. There is still a lot of mystery regarding the identity of the exogenous signal and its

978-1-7281-0404-1/19/$31.00 ©2019 IEEE
DOI 10.1109/iCMLDE.2018.00039

receptor, the critical gene and what it codes for: the Biochemical pathway triggered by the gene.

A theory I currently favor is that several exogenous factors control whether stem cell division is asymmetric or symmetric: (a) extra-cellular or cell adhesion molecules found in stem cell niches and (b) oxygen levels, for which the niche emerged by natural selection to control oxygen tension. A reason for (a) is that this critical gene must code for an extra-cellular protein, possibly a cell adhesion molecule in the stem cell niche. Why? Because when stem cells grow on feeder layers or irradiated stromal cells, they divide symmetrically, but when grown on specific extracellular matrices they can divide either symmetrically or asymmetrically pursuant of the nature of the matrix. Moreover, with respect to (b), all other factors held constant, if oxygen levels are low then stem cells divide symmetrically for longer, but if oxygen levels are high then stem cells divide asymmetrically.

More interestingly, if you put senesced cells on the extracellular matrix of young cells then they regain the ability to become non-senesced young cells, but if you instead place young cells on the extracellular matrix of senescent cells then they senesce. This may be explained as follows. When a stem cell sets down on its niche extracellular matrix it receives a signal via that interaction, setting off an intracellular pathway, a biochemical reaction, but it also receives signals from the physiological micro-environment: oxygen; amino acids; ions, vitamin growth factors; hormones; etc. Each of these signals sets off different intracellular signaling pathways. There must be an interaction between these and the pathway triggered by the extracellular matrix with a resulting sending of intracellular signals to the cell nucleus to turn on or off certain genes. This signaling also turns on and off the connexin or gap function genes, and concurrently also enacts either a vertical or horizontal plane of cell division.

This means that the cell which receives the signals corresponding to low oxygen and a given extracellular matrix in the niche now divides in a vertical plane and symmetrically. Both daughter cells are now attached to the matrix to receive the same extracellular signals. Both will have the exact same interior physiology, same gene expression and same methylation of DNA patterns. In the low oxygen "regime" and quiescent niche the stem cell will not experience gap junction communication.

A horizontal plane of cell division, however, is obtained when the cell is placed on a different extracellular niche matrix and it receives the extracellular signal corresponding to high oxygen or a specific growth factor. Now the mother cell which is attached to the niche extracellular matrix obtains as net effect of all these endogenous signals that it should remain a stem cell, but the daughter on top of her no longer receives the signal from the extracellular matrix, and its new micro-environment driven signals turn on the gap junction communication channels required for differentiation."

This knowledge from Biology inspires involving these three model elements towards Figure 1: (1) immutable DNA, (2) a cytoplasm context altered by the gap function communication and (3) the exogenous factor inputs.

The idea of the black stop cells of Figure 1 can be justified and understood with respect to a related Biological explanation, again from (James E. Trosko, personal communication, [2]):

Stem cells have only a few mitochondria, metabolizing glucose via glycolysis. As soon as they are exposed to oxygen, the Oct4 gene (the ultimate stem cell gene) is shut off and mitochondriogenesis occurs and the cell starts to metabolize glucose with abundant mitochondria via oxidative phosphorylation. It starts to multiply the mitochondria to the thousands, that is when it starts to lose its "stemness", starts to differentiate, and turns on its gap junction genes. It goes from a stem cell producing only a few ATP molecules from the glycolytic metabolism of glucose to producing about 17 ATP molecules per metabolism of glucose from oxidative phosphorylation. That means the stem cell has little excess energy to do anything besides staying alive, but the differentiated cell with tons of energy can use that energy to do many more adaptive tasks like make muscle, eye, brain, and liver cells.

When technology should become available to study the aforementioned transition from a low oxygen existing, quiescent stem cell with a few mitochondria, expressed Oct4 gene but no expressed gap junction gene, and only a few ATP molecules being produced during glycolysis, to a physiologically different state where high oxygen is present, oxidative phosphorylation is producing many ATP molecules in the presence of hundreds of mitochondria, no expressed Oct4 gene but expressed gap junction gene, then that is when one can identify the "new physiological state" that determines the new complex pathways between induction of intracellular signals that start to turn off and on genes, which, in turn, alters the physiological state of differentiated cells and the loss of "stemness". It would be the state by which the stem cell loses its "virginity" and, in my opinion, there is no reverting back to that virgin state.

When differentiated cells metabolize glucose via oxidative phosphorylation with their many mitochondria, they not only produce lots of ATP, but also many free radicals that can damage their cellular macromolecules, such as proteins, nucleic acids and membranes. That is why these differentiated cells usually have different kinds of protective mechanisms that salvage free radicals or to repair enzymes. However, even with these, the mitochondrial DNA is prone to mitochondrial (not genomic) DNA damage. If the cell losses one, two or a couple hundred mitochondria, it still can function and survive. When it loses too many, that is the signal for the cell to commit suicide or "apoptosis".

The simple model of Figure 1 could not only be further enhanced by a more faithful implementation of all of the above discussion but also, as already mentioned, by yet other thinking in Mathematical Biology [1] [3-4].

III. ALTERNATIVE MACHINE VISION EXAMPLE

The problem of machine vision considered is the problem of detecting an object of large variability and poor definition, a class of object. Let us look at an existing solution to this challenge [5-8] [12] and examine similarities and differences between it and a proposed bio-inspired method that draws upon the ideas of Figure 1.

As a post processor to a more standard object detection method, a computational ant explores the context of this detection to try to eliminate false alarms through exploration

of the context of the identification, e.g. which look like vehicle features but turn out to be pieces of the road or air vents on buildings as illustrated in Infrared IRLS sensor data from a Tornado fighter aircraft, Figure 2.

Figure 2 Multi-stage GP discovered vehicle detector [7]. Context becomes necessary to rule out false positives.

Starting at a suspected target, points in Figure 2, this computational ant meanders and at times becomes excited and deposits flags [5-6]. Figure 3 illustrates typical paths.

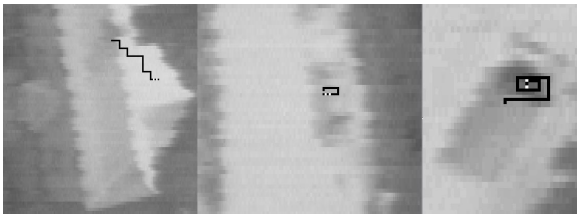

Figure 3 Paths taken by the computational ant [5-6].

An evolved function that takes for its input statistics about the distribution of these deposited flags (see [6] and RPB of Figure 4) uses this flag evidence to make the determination on whether the starting point was a genuine target or a false alarm.

How is this computational ant and flag placing algorithm discovered by GP? It was deliberately designed in what John Koza once described as "the principle of prospective analysis" [9]. That is, it enforces a helpful solution architecture. It is some way to incorporate analytical knowledge leaving GP to fill the details. Here, it implements a strategy reminiscent of foveation in animal vision.

Figure 4 reveals four branches to the ant algorithm, each evolving independently from an image truth to reject false alarms and retain true positives. The first branch, labelled M, moves the ant. It consults the second and third branch T? and F? which decide whether to turn the ant and whether to plant a flag at the location of the ant respectively. Once the ant ends its walk, statistics derived from the location of the flags are drawn and used in the result producing branch, labelled RPB. A positive evaluation of the RPB indicates the ant believes this to be an object, while a negative evaluation of the RPB means the starting point is a false alarm. The ant carries sensors that measure averages and standard deviations of pixel values in four directions and on concentric rings and textural statistics of a pixel area. Details can be found in [6].

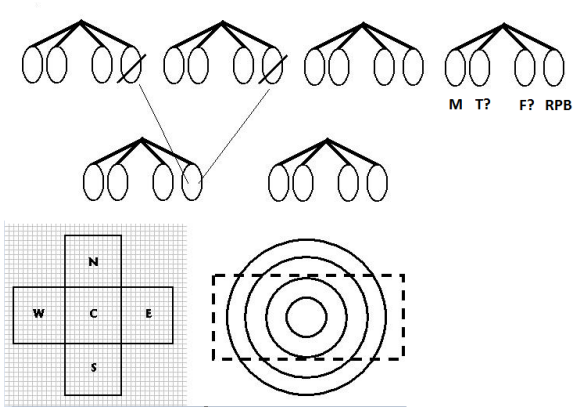

Figure 4 Genetic Programming implementation of ant.

IV. MODULARIZATION

GP does implement modularization and code re-use. For example, the M branch in Figure 4 explicitly re-uses or invokes the results of branches T? and F?: subroutines called by branch M. However, this re-use is dissimilar to that of Figure 1. What transpires in Figure 1 is that the same exact gene (all cells carry all genes) is fired in two cells at some distance from each other, thus producing symmetry, a type of modularized re-use, though not necessarily an identical but perhaps a congruent pattern, e.g., consider our human hands or ears, which are similar but indeed reflected.

Imagine a machine vision scheme where pattern re-use could be exploited in the detection of chairs from other objects in an image. Figure 5 shows some possible chairs.

Figure 5 e.g. chair concept to be detected in imagery

In the discussed GP machine vision [5-8] [12] scheme code gets reused in GP trees, or as a subroutine: Automatically Defined Function [10] or as a function sub-program or encapsulated subtree [11-12]. GP could in principle reuse the "leg" pattern but in practice, a uniquely representative module is seldom obtained by evolution that corresponds to the sub-object [11]. Dividing runs into explicit stages [7], resembling bagging and boosting prior to the popularity of such nomenclature, does achieve a type of sub-object identity as in Figure 6, but not as a module, as here, multirun GP drives unspecific data splitting [7-8].

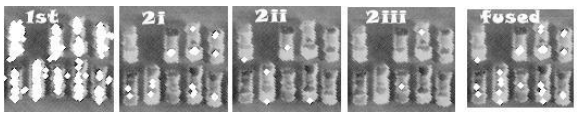

Figure 6 Vehicle detection with separate runs of GP [7].

V. SUGGESTED SCHEME: EVOMORPH

GP can implement modularization and code re-use. For example, M branch in Figure 4 explicitly re-uses/invokes the output of branches T? and F?: subroutines that it calls. Yet, Figure 1 suggests an alternative algorithm, a powerful scheme which reminds of a function-like flexible image morphological template matching-like scheme endowed with pixel location-controlled modularization. It could also enjoy other Mathematical Biology properties [1][3-4]. It is proposed for the first time here as seed for further research.

As with the computational ant scheme, a starting point is needed and, as in the ant scheme, this is taken from a pre-processor - a cruder quicker detector that also produces false alarms. The idea is to use GP to evolve: (1) the DNA (a memory); (2) the makeup of the "chemistry" in the progenitor (starting) cell (a computer program); (3) the rate of transmission of "chemistry" from mother to daughter via gap functions (computer program with a memory that accounts for cell generational distance).; (4) the function that takes this intracellular chemistry and exogenous inputs (pixel values of the image sensed) to fire the gene that tells how to divide and when to slow down or stop. (5) that function that takes all inputs to compute a small template at cell location. Once the organism is built, templates are checked against each localized pixel-based statistics. GP fitness measures the degree of overall template matching. If matched considerably then the object is detected else not.

Consider Figure 7. The scheme needs to detect "chair" as distinct from everything else, for example, a chandelier.

Figure 7 Detect the concept chair as distinct from other objects such as chandelier.

Cells should be bigger than m pixels, e.g. m = 20, because pixel jumps are necessary for speed and because template matching must work on a pixel patch. Figure 8 shows an evaluation in progress. It may be that the template matches the image exactly or not. Even so, the fitness is driven by the evaluation of the template matching! The scheme should work learning to detect an object class of large variability and of poor definition such as "chair" as distinct from everything else. It may prove sufficiently powerful to also discover the partially occluded object.

Finally, a big challenge will be to decide where to place the progenitor cell? This may be achieved with a multi-resolution strategy, an exploratory strategy that seeks a unique feature typical of the object as the starting point, or a strategy that grows more than one organism at different proximate locations. Considerable research effort will need to go into this aspect of implementation.

ACKNOWLEDGEMENTS

The author is grateful to Professor James E. Trosko for generously communicating his wide and in-depth knowledge of the subject of stem cells, and to Mark Edwards of Worcester, UK for many conversations.

Figure 8 Evomorph growing the organism.

REFERENCES

[1] Turing, Alan M. (1952). "The Chemical Basis of Morphogenesis". Philosophical Transactions of the Royal Society of London B. 237 (641). pp. 37–72.

[2] Private Communiations (2017): Trosko, James E., Professor Emeritus, Michigan State University.

[3] Zeeman E.C. (1974), "Primary and secondary waves in developmental biology", AAAS 1974, Some Mathematical Questions in Biology VIII, Lectures on Mathematics in he Life Sciences 7, Amer. Math. Soc., Providence, R.I. 1974, 69-161.

[4] Cooke J, Zeeman EC (1976), "A clock and wavefront model for control of the number of repeated structures during animal morphogenesis", J Theor Biol. 1976 May 21; 58(2): 455-76.

[5] Howard D. (2003), "Innovating with Automatic Programming". Journal of Defence Science, 8(2):76-82, 2003.

[6] Howard D., Roberts S. C. et al. (2006), "Pragmatic Genetic Programming strategy for the problem of vehicle detection in airborne reconnaissance". Pattern Recognition Letters, 27(11):1275-1288, 2006.

[7] Roberts S.C., Howard. D. (1999), "Evolution of Vehicle Detectors for Infrared Linescan Imagery". In Riccardo Poli and Hans-Michael Voigt and Stefano Cagnoni and Dave Corne and George D. Smith and Terence C. Fogarty editors, Evolutionary Image Analysis, Signal Processing and Telecommunications: First European Workshop, EvoIASP'99 and EuroEcTel'99, volume 1596, pages 110-125, Goteborg, Sweden, 1999. Springer-Verlag.

[8] Howard D., Roberts S.C., Brankin R. (1999), "Target detection in SAR imagery by genetic programming". Advances in Engineering Software, 30(5):303-311, 1999.

[9] Koza J. (1992), "Genetic Programming: On the Programming of Computers by Means of Natural Selection: v. 1 (Complex Adaptive Systems) ", MIT Press.

[10] Koza J. (1994), "Genetic Programming II: Automatic Discovery of Reusable Subprograms", MIT Press.

[11] Roberts S.C., Howard D., Koza J.R. (2001), "Subtree encapsulation versus ADFs in GP for parity problems". In Lee Spector and Erik D. Goodman and Annie Wu and W. B. Langdon and Hans-Michael Voigt and Mitsuo Gen and Sandip Sen and Marco Dorigo and Shahram Pezeshk and Max H. Garzon and Edmund Burke editors, Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001), page 186, San Francisco, 2001.

[12] Roberts S.C., Howard D., Koza J.R. (2001), "Evolving modules in Genetic Programming by subtree encapsulation". In Julian F. Miller and Marco Tomassini and Pier Luca Lanzi and Conor Ryan and Andrea G. B. Tettamanzi and William B. Langdon editors, Genetic Programming, Proceedings of EuroGP'2001, volume 2038, pages 160-175, Lake Como, Italy, 2001. Springer-Verlag.