

Modeling Hierarchical Architectures with Genetic Programming and Neuroscience Knowledge for Image Classification Through Inferential Knowledge



Gustavo Olague, Matthieu Olague, Gerardo Ibarra-Vazquez, Isnardo Reducindo, Aaron Barrera, Axel Martinez, and Jose Luis Briseño

Abstract Brain programming is a methodology based on the idea that templates are necessary to describe artificial dorsal and ventral streams and their combination into an artificial visual cortex. We review the main concerns by introducing some initial thoughts about the status of genetic programming and other methodologies related to our research work. This chapter proposes the hierarchical integration of two architectures (templates) to enhance the quality of acquiring artificial visual percepts. We theoretically justified the necessity for designing manual hierarchical architectures. Planning complex structures through inferential knowledge simplify the design while adopting current technology. The methodology base its analysis on providing domain knowledge (neuroscience) at a higher level while looking for better computational structures within a local (lower) level. The efficiency of searching for optimal

G. Olague (✉) · A. Barrera · A. Martinez · J. L. Briseño
EvoVisión Laboratory, CICESE, Ensenada, B.C. 22870, Mexico
e-mail: olague@cicese.mx

A. Barrera
e-mail: abarrera@cicese.edu.mx

A. Martinez
e-mail: amartinez@cicese.edu.mx

J. L. Briseño
e-mail: brisenoc@cicese.mx

M. Olague
Anahuac University Queretaro, 76246 El Márques, Querétaro, Mexico
e-mail: matthieu.olague03@anahuac.mx

G. Ibarra-Vazquez
ITESM, Institute for Future of Education, Monterrey, N.L., 64849, Mexico

I. Reducindo
Autonomous University of San Luis Potosí, Information Sciences Faculty, Fracc. Talleres, San Luis Potosí 78494, Mexico
e-mail: isnardo.reducindo@uaslp.mx

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023
L. Trujillo et al. (eds.), *Genetic Programming Theory and Practice XIX*,
Genetic and Evolutionary Computation, https://doi.org/10.1007/978-981-19-8460-0_7

architectural configurations proceeds from deductive and inductive reasoning. This chapter brings a proposal of abductive reasoning to enrich the brain programming paradigm by taking advantage of computational re-use of dorsal stream discoveries while enhancing the overall complexity of the final proposal. We propose a Visual Turing test to establish the quality of the proposal in comparison with the state of the art. The results show that our methodology can produce consistent outcomes during training and testing, representing significant progress toward thought representation.

1 Introduction

This chapter studies artificial intelligence (AI) from symbolic learning, considering image classification problems. Symbolic learning is one of the classical approaches to AI, whose development has been complex regarding image classification due in part to the lack of a theory to create sophisticated representations. Symbolic learning is associated with identifying mathematical expressions trained with sets of images. The study of symbolic learning for image classification through genetic programming (GP) framework currently lacks a transparent methodology able to beat other state-of-the-art approaches [11].

This document introduces the idea of abductive reasoning in GP considering hierarchical structures inspired by the human visual cortex. We study the problem of symbolic representation under image classification following a variant of the Turing test, which is a method of inquiry in AI for determining or not if a computer is capable of thinking. The original test does not require that the machine intelligence replicates thinking like a human being, only that a computer could trick a human into believing that the machine is human. Nonetheless, the seminal test requires the device to exhibit some understanding of natural language processing (NLP). In our case, we apply a visual Turing test based only on identifying visual information unrelated to language use but on understanding visual concepts, i.e., the identification and association of visual stimulus with emotion recognition.

To outline an approach departing from current machine learning (ML), we took ideas from inferential knowledge; see Sect. 3. Our strategy identifies different types of reasoning susceptible to being transcribed into the form of computer algorithms. The methodology follows conceptual representation in deductive, inductive, abductive, retroductive, and transductive reasoning. With these elements, we introduce the first proposal of abductive logic based on the results of previous research through the combination of deductive and inductive methodologies. The association of earlier studies regarding salient object detection and image classification influences the proposal's value since the new plan significantly increases the size of solutions while helping to find answers to more complex problems and taking advantage of the program searches made in previous studies.

Nowadays, the size of computer solutions to computer vision (CV) problems like image classification reaches unique requirements that digital companies can only achieve many times. For example, Google trained in 2021 a two billion parameter

AI vision model on three billion images while achieving 90.45% top-1 accuracy on ImageNet. Also, in 2021, Facebook implemented a SELF-supERvised (SEER) approach for CV inspired by NLP with billion parameters to learn models from any group of images on the internet. Indeed, regarding NLP, the Generative Pre-trained Transformer 3 (GPT-3) has 175 billion parameters, and basically, the architecture design of all these networks follows a handcrafted approach. This situation makes us think about the unsuitability of searching for optimal architecture design through genetic and evolutionary computation (EC) approaches, at least with currently available technology.

We pause our discussion in Sect. 4.1 to a theoretical biological problem related to the Word Links game to illustrate the problem of blindly attempting to search for a solution from scratch. The frequency and distribution of functional amino acid sequences result in colossal search spaces, making it impossible to automate the search. This situation helps us to introduce our approach to designing symbolic representations through what we call templates inspired by classical neuroscientific representations that are not based on neural networks but rest on mathematical modeling. It is possible to extrapolate the template idea to neural architecture search.

Moreover, as we will explain here, the images have the constraint that the learned concepts must be contingent. Programming a computer to represent a contingent thought implies that the concept may be true or false, whereas non-contingent thoughts are necessarily true or false. In other words, databases portraying concepts like ImageNet contain empirical knowledge, while databases characterized by psychological information like OASIS include information that can be known through reason, by analysis of concepts, and by valid inferences.

GP is a technique for automatic programming (AP) centered on the problem of machine learning. The idea is to automatically build computer programs to produce some machine intelligence by approximately solving challenging computational tasks. In this way, our research touches on several areas: automatic programming, machine learning, computer vision, artificial intelligence, and evolutionary computation.

Exploring the opportunities and limitations of current genetic programming and related research areas help in understanding the reasons behind past failures and future success. Hence, we propose to review the subject since it has roots in several myths and open issues around automatic programming, genetic programming, and mainstream research on computer vision and deep learning (DL).

2 Myths and Prospects of Genetic Programming

Scientists and Engineers understand automatic programming as computer programming where machines generate code under certain specifications. Translators could be considered automatic programs mapping high-level language to lower-level language, i.e., compilers. Automatic programming includes applying standard libraries into what is known as generative programming, so the programmer does not need to

Table 1 Assessment of myths on automatic programming in other research areas

Myth	Method		
	Machine learning	Computer vision	Evolutionary computation
(a)	✓	✗	✓
(b)	✓	✗	✓
(c)	✗	✗	✗

re-implement or even know how some piece of code works. Also, the idea of creating source code based on templates or models based on a visually graphic interface where the programmer is instead a designer who, through drag and drop functions, defines how the app works without ever typing any lines of code. In 1988, Rich and Waters identified some myths and realities about automatic programming, which are still relevant today [23]. Next, we present a list of the relevant points helpful to our exposition:

- (a) The myth implies that automatic programming does not need domain knowledge.
- (b) The myth is that general purpose and fully automatic programming are possible.
- (c) The myth indicates that there will be no more programming.

We could extrapolate these three myths into ML, CV, and EC to give a snapshot of their actuality (Table 1). Regarding ML, the myths of unnecessary domain knowledge, general purpose, and fully automatic programming are at the methodology's core, while the third myth is not a pursued property. For CV, none of the myths apply since this discipline centers on the study of vision to recreate visual perception on a problem-by-problem basis. In the case of EC, we found the same situation concerning ML. Nevertheless, genetic programming as an approach aiming at AP, we consider that all myths apply in general.

GP is a research paradigm driven by the idea “tell the machine what to do, not how to do it”, as explained by Koza [12] while paraphrasing the words of Samuel [27]. The methodology centers on the goal of accomplishing program induction as the process for building complete working programs. Nevertheless, the solutions miss the property of scalability. A pervasive myth anchored on the idea that if we want to achieve complete automatic programming, GP requires only general principles and no human assistance in such a way of being domain independent. Paradigmatically, the kind of solutions is usually narrow and specific [13]. Indeed, Koza predicted in 2010 that GP would enhance due to Moore's law or the increased availability of computer power—a situation that did not arrive. This chapter presents the idea that to manage an ever-increasing problem; we need a way to enrich the problem representation paired with a scheme that processes inferential knowledge. O'Neill and Spector equate the challenge of achieving automatic programming with reaching a complete solution to the AI problem [22]. We propose a novel visual Turing test as an instance of an unsolved problem that can drive us in the long term to study the scheme introduced in this chapter.

We believe that many authors recognize the issues and implications of following the myths and hide part of their approach since this will be problematic at the moment of submitting their work for peer review or simply because they are so involved with the paradigm that they do not stop to report the other methods used in the investigation. Also, many people working on GP study the methodology as part of ML without looking into AP, while the opposite is rare. We believe that GP encompasses both research areas and considers it when developing solutions to CV problems. Indeed, it is unrealistic to expect that GP alone will be sufficient to achieve automatic programming, and researchers usually incorporate other methods without correctly reporting them in the manuscripts. We do not claim that the scheme reported here is novel and left other researchers the opportunity to clarify how they approached their work according to the ideas presented in this chapter. Nevertheless, the innovative representation follows mainstream cognitive science combined with GP to create adaptatively complex models of visual streams to approach salient object detection and image classification problems. Also, our contribution roots in the scheme application to architectural representations and how to increase solutions complexity through the hierarchical combination of architectures.

In 2010, O'Neill et al. recognized several open issues that people working on GP need to address if further development of the area is achievable in such a way to realize the full potential and become conventional in the computational problem-solving toolkit [21]. Next, we recall the main open issues relevant to our discussion:

- (a) GP representation, modularity, complexity, and scalability. The topic relates to identifying appropriate (optimal) representations for GP while considering structure adaptation based on some measure of quality that captures the relationship between the fitness landscape and the search process. Also, the idea relates to the tenet that the invention of accompanying modularity and complexity of function at a high level originates from the development or growth of simple building blocks. Regarding scalability or the ability to provide algorithmic solutions to problems of substantial size/dimensionality, the authors recognize the difficulty since the GP process always tends to compromise between the goals set by the fitness function and the program's complexity. Authors claim that GP's strength resides in providing a natural engine for generalization, whose obstacles can be overcome only with recipes for modularity and scalability.
- (b) Domain knowledge. This subject refers to the idea of defining an appropriate AI ratio. Koza highlighted that for a minimum amount of domain knowledge supplied by the user (the intelligence), GP achieves a high return (the artificial). All this situation is about attaining human-competitive results automatically and partly measured by the existence of a high AI ratio.
- (c) GP benchmarks and problem difficulty. This issue relates to the problem of determining a set of test problems that the scientific community can rigorously evaluate, and as a result, the algorithmic discoveries can be accepted based on such benchmarks.
- (d) GP generalization, robustness, and code re-use. These elements refer to the nature of program representation and the qualities the designer attempts to fulfill. As

stated by the authors, GP generalization relates to ML and the idea of overfitting as in statistical analysis; therefore, such concepts relate to the robustness of solutions and code re-use.

According to the authors listing the above open issues, GP was not universally recognized as mainstream and trusted problem-solving strategy despite numerous examples where the technique outperforms other ML methods. They argue that the resistance of the ML community to embrace GP is rooted in Darwinian thinking. Instead, we believe it is more related to the overuse of random processes and the implicit connection taken for granted that structures raise from fitness. The second statement is made explicit by Koza in such a way that the GP paradigm address the problem of getting computers to learn to program themselves by providing a domain-independent way to search the space of possible computer programs for a program that solves a given problem. This idea resides on the derivation of knowledge following a way to do program induction. In the words of Koza:

“...That is, genetic programming can search the space of possible computer programs for an individual computer program that is highly fit in solving (or approximately solving) the problem at hand. The computer program (i.e., structure) that emerges from the genetic programming paradigm is a consequence of fitness. That is fitness begets the needed program structure.” (Koza, 1992, p. 3)

After carefully reading the O’Neill et al. article, we assume that such principle anchors in the imaginary and that designers believe it is true-valid when designing a search procedure for the artificial induction of programs. As we will propose in this chapter, we believe that to achieve program synthesis; we need a way of replicating the theory of inferential knowledge in the sense that to advance the search for complex programs when attempting to solve challenging problems, we need to recognize not only the idea of induction, but also other ways of inferential knowledge like deductive reasoning. The whole scientific approach requires a complete cycle of deductive–inductive reasoning. We understand that this idea of structure raising from fitness is a myth like those identified in automatic programming.

Again we can extrapolate the identified open issues to observe their relevance in the proposed three research areas (Table 2). Representation, modularity, complexity, and scalability are open issues in ML, CV, and EC. On the other hand, generalization, robustness, and code re-use are well-studied subjects with clear theoretical and practical guides in all three research areas. Regarding domain knowledge, ML and CV incorporate it systematically as part of their approach to problem-solving; nonetheless, EC follows a similar path of leaving the charge of building intelligent solutions from apparently disconnected building blocks based on the search process. Finally, benchmarks and problem difficulty are pervasive problems in ML and EC since both attempt to discover algorithms that can be useful in an extensive range of

Table 2 Assessment of GP—open issues in other research areas

Myth	Method		
	Machine learning	Computer vision	Evolutionary computation
(a)	✓	✓	✓
(b)	✗	✗	✓
(c)	✓	✗	✓
(d)	✗	✗	✗

different problems, which is not the case in CV, where each problem follows strict formulations.

Nowadays, the situation is still more worrisome with the worldwide impact of DL since this technology is not even contrasted with alternative formulations because most people in computer science estimate that there is no competitive alternative. However, such a paradigm has numerous sometimes deadly failures [3], and since overcoming such hurdles is not an easy task, the always present possibility of lack of success represents an opportunity for other research paradigms [11]. Here, we recall some aspects where neural networks’ lack of success impact trust and confidence in these systems:

- (a) **Brittleness and lack of trustworthiness.** These aspects refer to the lack of invariance whose attempts to overcome lay on extending the number of patterns, including rotation, scaling, illumination, and as many as different representations of a kind of object. Nevertheless, input data corruption through adversarial attacks is a severe source of difficulty. Moreover, contamination with information in the form of stickers or other ways of marking traffic signals, cars, persons, or other objects represents real conundrums for such technology.
- (b) **Uncertainty.** Classification systems based on DL report high certainty for the training data, but the way of calculating robustness is still an open issue. Data-dependent methodologies need improvement when calculating and dealing with uncertainty toward a robust approach whose accuracy refines its confidence in the outcome.
- (c) **Data-driven technology and expensive hardware requirements.** One of the most significant disadvantages of deep learning is that it requires considerable data to perform better than other techniques. It is costly to train due to complex data models. Thus, this technology requires expensive Graphic Processing Units (GPUs) and hundreds of machines. This data dependency is the source of most of the problems described in this section.
- (d) **Embedded bias and catastrophic forgetting.** Bias is the Achilles heel of data analysis, divided into data bias and bias in evaluating or creating data. DL as ML depends on accurate, clean, and well-labeled training data to learn from so they can produce accurate solutions. Most AI projects rely on data collection, cleaning, preparation, and labeling steps, which are sources of bias or catastrophic

Table 3 Assessment of DL—lack of success in other research areas

Myth	Method		
	Machine learning	Computer vision	Evolutionary computation
(a)	✗	✗	✓
(b)	✗	✗	✓
(c)	✓	✗	✓
(d)	✗	✗	✗
(e)	✗	✗	✓

forgetting. This last issue results from updating data, like in the case of detecting artificially generated fake images. In order to repair the performance of CNNs, the researchers trained them with new data, and as the cycle continued, the CNN forgot how to detect the old ones, even the original clean data.

- (e) Explainability. The advantage of an end-to-end methodology combined with the black-box paradigm makes it hard to conclude why the system selects certain features, especially for problems with many classes or when different networks provide solutions to the same problem. How DL reaches conclusions is a mysterious process that is somewhat stuck on the theoretical side and left to designing better, more constrained datasets.

Table 3 summarizes the analysis of these issues in the other three research areas that can complete our critique of the situation in optimization and learning approaches. Brittleness and lack of trustworthiness are not issues for ML and CV, but these are problems for EC. Researchers working in ML and CV do not consider EC as part of their popular methodologies; therefore, it is necessary to work on these aspects to achieve widespread acceptability. Uncertainty and explainability are also red flags for EC, while both are well-studied in ML and CV. In the case of data-driven technology and expensive hardware requirements, ML and EC have issues, while CV has no problems. Finally, none of the other research areas has bias and catastrophic forgetting issues.

Next, we propose to apply some strategies founded on inferential logic for building knowledge to enhance designed models to approach more challenging problems. Thus, we start to explore the scheme proposed in this chapter as a way to create complex solutions based on hierarchical architectures of the visual stream.

3 Inferential Knowledge

There are four types of knowledge representation in AI: relational knowledge, inheritable knowledge, inferential knowledge, and procedural knowledge [14]. The first refers to storing facts, and it finds application in relational database systems. The

second deals with ways to represent data with some hierarchy, subclasses inherited from the superclasses in the representation—the superclass holds all the data in the subclasses. The third is essential for our discussion since inferential knowledge defines understanding in terms of formal logic conditions and has a strict rule. The knowledge raises from objects by studying their relation. Finally, procedural knowledge in AI represents control information in small programs and codes to describe how to proceed and do specific tasks, i.e., usually if-then rules. Paradigmatically, this is what we need to answer Samuel’s question, but first, we need an approach to create knowledge.

Knowledge representation needs a set of strict rules which can derive more facts, verify new statements, and ensure correctness. Many inference procedures are available from several types of reasoning. The reasoning is part of intelligence, and to derive artificial ways of creating knowledge, we first need to understand how humans abstract it. Even though humans use several ways of creating knowledge, let us focus on logic first. In logic, inference refers to a process of deriving logical conclusions from premises known or assumed to be true.

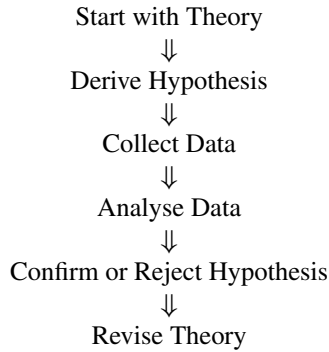
An inference is valid if based upon sound evidence and the conclusions follow logically from the premises [16]. The reasoning divides into deductive, inductive, analogical, abductive, cause-and-effect, decompositional, and critical thinking. The cycle of deductive–inductive reasoning/theory as scientific explanation drives the overall conception of science. Theoretical systems like Newton’s laws and Kepler’s laws show us how to interpret scientific theory in terms of explanations. Examples of modeling visual information to accurately predict complex corners and retro-reflective targets in terms of morphology, geometry, and physics is made with regression following deductive reasoning despite application of optimization techniques [17]. Next, we focus on a typical division made of the following five types of reasoning.

3.1 Deductive Reasoning

This kind of reasoning helps make predictions, which contributes to knowledge. The reasoning starts with ideas to generate hypotheses to produce observations.

Testing the Theory → Observations/Findings

The hypothesis is a supposition, preposition, or principle that is supposed or taken for granted to draw a conclusion or inference for proof of the point in question. For deductive reasoning to work, hypotheses must be correct since the conclusion follows with certainty from the premises. Thus, the hypotheses follow examinations to derive logical conclusions—this is true for all members. Therefore, we achieve a theoretical explanation of what we have observed, which is the contribution to knowledge.



This is summarized through the following relationship:

$$\underbrace{\mathcal{A}}_{\text{(Rule)}} + \underbrace{\mathcal{B}}_{\text{(Case)}} = \underbrace{\mathcal{C}}_{\text{(Result)}} \quad (1)$$

and is exemplified with the following syllogism:

$$\begin{array}{l}
 \text{(Deductive Reasoning)} \quad \begin{array}{l} \text{All men are mortal;} \\ \text{but Seth is a Man.} \end{array} \\
 \hline
 \text{Therefore Seth is a mortal.} \quad \therefore
 \end{array}$$

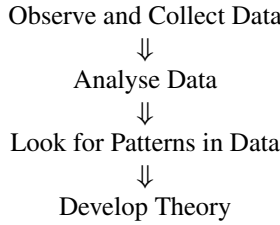
This way of approaching problems is classical of CV and is the method we use in EvoVisión as explained in [17]. Nevertheless, the following method is classical to GP, and this chapter attempts to expose the differences and complementarities to adopt more complex methodologies for problem-solving.

3.2 Inductive Reasoning

Inductive research begins with a research question and empirical data collected to generate a hypothesis and theory.

Observations/Findings \rightarrow Testing the Theory

It is the inverse of the deductive approach. Inductive learning starts with the phenomena, and the researcher needs to be careful about the inferred rules since these are based on observations, not speculations. The observations produce patterns, and we obtain rules or theories from these. Therefore, the conclusion follows not with certainty but only with some probability.



This is summarized through the following relationship:

$$\underbrace{A}_{\text{(Result)}} + \underbrace{B}_{\text{(Case)}} = \underbrace{C}_{\text{(Rule)}} \tag{2}$$

and is exemplified with the following syllogism:

(Inductive Reasoning) $\frac{\text{Seth is a mortal}; \text{ but Seth is a Man.}}{\text{Therefore All men are mortal.} \quad \therefore}$

nevertheless, the following syllogism represents an unsound argument

(Inductive Reasoning) $\frac{\text{Seth eats meat}; \text{ but Seth is a Man.}}{\text{Therefore All men eat meat.} \quad \therefore}$

The researcher must ensure that all generalizations are in the theory or rules since the contribution to knowledge should be routed in a sound argument.

3.3 *Abductive Reasoning*

Note the same construct in a different order; therefore, in inferential knowledge, we get different research-design approaches. This process helps derive the contribution to knowledge. In the abductive approach, we must provide the best possible explanation for what we have observed, even if the observation is incomplete. Note that this does not refer to gathering incomplete data and calls it abductive. The best explanation of incomplete observation uses rules plus results. We obtain an explanation from observations (unexpected observation—little surprise) and an idea (rules or theories). This approach is practical for testing hypotheses (similarities with the process of generalization or transfer learning).

This is summarized through the following relationship:

$$\underbrace{A}_{\text{(Rule)}} + \underbrace{B}_{\text{(Result)}} = \underbrace{C}_{\text{(Case)}} \tag{3}$$

and is exemplified with the following syllogism:

$$\begin{array}{l}
 \text{(Abductive Reasoning)} \quad \begin{array}{l} \text{All men are mortal;} \\ \text{but Seth is a mortal.} \end{array} \\
 \hline
 \text{Therefore Seth is a Man.} \quad \therefore
 \end{array}$$

3.4 *Retroductive Reasoning*

The retroductive approach aims to understand why things are the way they are.

This is summarized through the following relationship:

$$\underbrace{A}_{\text{(Result)}} + \underbrace{B}_{\text{(Cause)}} = \underbrace{C}_{\text{(Cause)}} \quad (4)$$

and is exemplified with the following syllogism:

$$\begin{array}{l}
 \text{(Retroductive Reasoning)} \quad \begin{array}{l} \text{Seth is a mortal;} \\ \text{but Seth is dead.} \end{array} \\
 \hline
 \text{Therefore What killed Seth.} \quad \therefore
 \end{array}$$

3.5 *Transductive Reasoning*

This is summarized through the following relationship:

$$\underbrace{A}_{\text{(Result)}} + \underbrace{B}_{\text{(Cause)}} = \underbrace{C}_{\text{(Cause)}} \quad (5)$$

and is exemplified with the following syllogism:

$$\begin{array}{l}
 \text{(Transductive Reasoning)} \quad \begin{array}{l} \text{Seth is a mortal;} \\ \text{but What killed Seth?} \end{array} \\
 \hline
 \text{Therefore Seth is dead.} \quad \therefore
 \end{array}$$

4 Inferential Knowledge in Brain Programming

Brain programming (BP) is a research paradigm based on fusing cognitive computational models used as templates with a powerful search mechanism to discover symbolic substructures embedded within the higher graph structure. The first results were reported in 2012 when an artificial ventral stream was proposed to approach an object recognition problem [4]. The task consists of identifying critical features (interest region detection and feature description) of an artificial “what” pathway to simplify the whole information process, see Fig. 1a. The strategy reduces the total

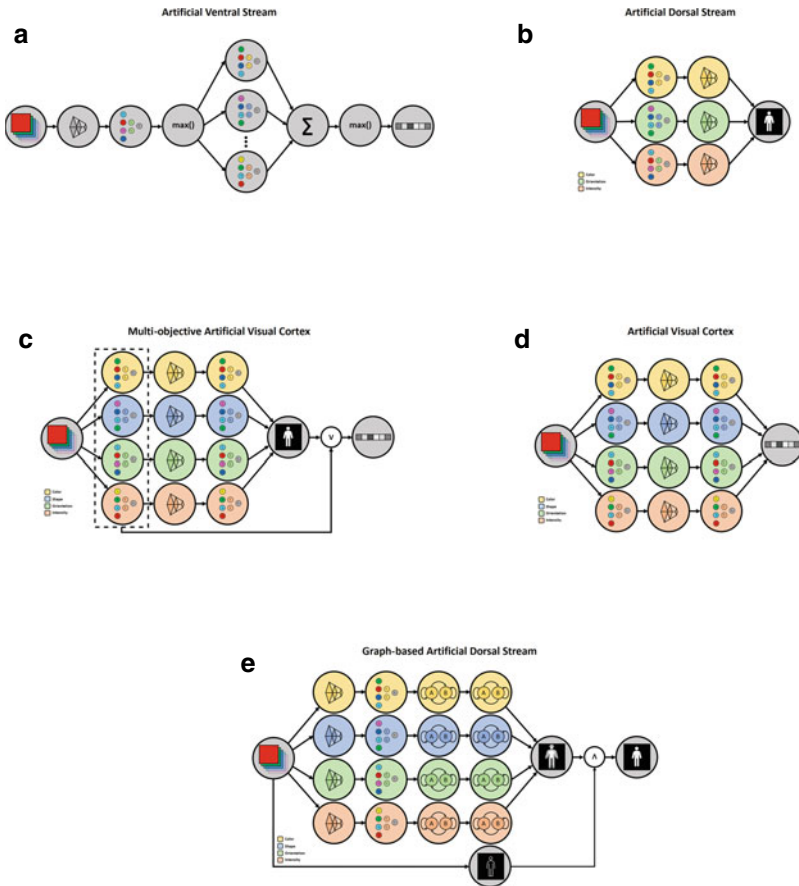


Fig. 1 Brain programming applies a template to characterize the behavior that the designer is attempting to recreate in the algorithm. Here, we show five different algorithms that we reported previously. **a** Object Recognition with AVS [4]. **b** Visual Attention with ADS [6]. **c** Visual Attention plus Object Recognition [9]. **d** Object Recognition with AVC [18]. **e** Graph-based Visual Attention [20]

computational cost by substituting a set of patches with an offline learning process to enforce a functional approach. Later, BP improved an artificial dorsal stream by searching for optimal programs embedded within a visual attention architecture, see Fig. 1b. The process incorporated learning to a handmade technique primarily based on deductive and heuristic reasoning [6].

BP design improved with the idea of studying two tasks simultaneously (salient object detection and image classification) and applying the framework of multi-objective optimization [9]. The design incorporates the V1 stage of visual attention into the artificial ventral stream to create a kind of artificial visual cortex, see Fig. 1c. The design focuses on the computation of the feature descriptor into a particular image region. The results significantly improved the performance of previous ventral stream processing.

An enigmatic result of the artificial visual cortex consists of the random discovery of perfect solutions for non-trivial object recognition tasks [18]. The design incorporates the V1 stage of visual attention and the idea of parallel computation across different visual dimensions, see Fig. 1d. The design considers integrating four visual dimensions after transformations with a set of visual operators discovered by multi-tree GP.

Recently, the EvoVisión laboratory proposed to evolve an ADS following the idea of graph-based visual attention while incorporating the dimension of form and evolving multiple functions with a multi-tree GP [20]. The new algorithm includes an image segmentation process merged with the output of the visual saliency process to create the proto-object, see Fig. 1e.

4.1 Doublets

Lewis Carroll proposed a game called “Word Links” around Christmas 1877 as a form of entertainment for two bored young ladies. The game is a kind of puzzle whose rules Lewis submitted to the journal *Vanity Fair* and published in March 1879.

The pastime consists of proposing two words of the same length, and the puzzle consists in linking these together by interposing other words, each of which shall differ from the next word in one letter only. The puzzle starts with the two words, called doublets,¹ and the player looks for interposing words, called links, by changing one letter from the given words until finishing connecting both words. The entire series, called a chain, from which an example is here:

¹ In linguistics-etymology, people call two or more words in the same language doublets or etymological twins or twinlings when they have different phonological forms but the same etymological root.

HEAD
 heal
 teal
 tell
 tall
 TAIL

What is important to understand is that according to TWL (Tournament Word List) scrabble dictionary, there are 4214 four-letter words with meaning, while there are $26^4 = 456,976$ possible combinations of words from the alphabet. Each and every four-letter word has $25 \times 4 = 100$ neighboring sequences, but only $4214 \times 100 / 456976 = 0.92$ neighbors, on average, have meaning. This game illustrates a complex system where meaning is not randomly distributed [8].

This popular word game serves as an analogy to understand the problem of frequency and distribution of amino acid sequences which are functional, either as enzymes or in some other way. The alteration of a single letter corresponds to the most straightforward evolutionary step, the substitution of one amino acid for another, and the requirement of meaning corresponds to the requirement that each unit step in evolution should be from one functional protein to another [15]. If natural selection evolves, functional proteins must form a continuous network (series of small isolated islands in a sea of nonsense sequences) that can be traversed by unit mutational steps without passing through non-functional intermediates. Salisbury calculates the search space for a small protein as follows:

A typical small protein contains about 300 amino acids, and its controlling gene about 1,000 nucleotides (three for each amino acid). Because each nucleotide in a chain represents one of four possibilities, the number of different kinds of chains is equal to the number 4 to the power of the number of links in the chain; that is, $4^{1,000}$, or about 10^{600} . (Salisbury 1969, p. 342)

Then, Salisbury imagined a primeval ocean uniformly 2km deep, covering the entire Earth, containing DNA (deoxyribonucleic acid) at an average concentration and each double-stranded molecule with 1,000 nucleotide pairs. Also, he imagined each DNA molecule reproducing itself one million times per second and occurring a single mutation each time a molecule reproduces, while no two DNA molecules are ever alike [26]. Salisbury estimates that in four billion years, the production is about 7.74×10^{64} different kinds of DNA molecules, and if we consider 10^{20} similar planets in the Universe, we obtain a total of 7.74×10^{84} (roughly 10^{85}) different molecules. So he concludes: "If only one DNA molecule were suitable for our act of natural selection, the chances of producing it in these conditions are $10^{85} / 10^{600}$ or only 10^{-515} ". The size of these numbers receives the name hyper astronomical, and the probability of occurring makes us think of an impossibility.

Regarding brain complexity, researchers base brain organization on regions. Within computer science, it is popular to think of it in terms of neural mechanisms,

although other authors explain the evolutionary development based on volume and mass [10]. The human brain contains about 100 billion neurons, more than 100,000 km of interconnections, and has an estimated storage capacity of 1.25×10^{12} bytes. This is equivalent to 10^{15} connections and contains roughly the same number of neurons as there are stars in the Milky Way. The analogy is with an analog device where the explanation of information flow depends on synaptic processing time, conduction speed, pulse width, and neuron density. In other words, the understanding of the brain depends on modeling the information processing capability per unit time of a typical human brain as a function of interconnectivity and axonal conduction speed. Indeed, such an analogy helps create numerical models/methods at the forefront of technological advances. However, a missing part is an analogy with information processing devices that manage language and code like in the doublets example and the DNA. Brain programming proposes to create an analogy with an information process based on symbolic computation. Nowadays, current computational technology is meager compared to small parts of a brain region and understanding how language could be possible within neurons. The idea is to adapt the current (handmade) cognitive computational proposal with GP to incorporate learning. In this chapter, we extended GP beyond purely inductive reasoning while recognizing the need for a deductive approach and the first proposal of abductive learning.

4.2 *The Visual Turing Test*

Nowadays, thousands of researchers see DL as the holy grail to approach AI since this technique proved that it is possible to solve previously thought challenging problems while ignoring the risks they may pose. As we reviewed in Sect. 1, the GP open issues require a benchmark with a proven problem difficulty that involves domain knowledge and complex representation. The aim is to force the designer on many axes about modularity, scalability, generalization, robustness, and code re-use. Indeed, as a powerful technique, DL moved the limits of AI and launched society into a new stage of computational development. However, as the AI frontier erodes, it is necessary to create new benchmarks that help us to define in practice what we mean by intelligence. The idea of intelligence is not new, and many believe, at least in AI, that Turing adequately defined the term in his now famous test [28]. Nevertheless, the idea has origins that date back over two millennia and are in the Talmud-Sanhedrin 65b:

Rava created a person, and sent it before Rav Zeira
Rav Zeira spoke to it, but it would not reply
Rav Zeira said to it
you are a creation of one of the magicians, return to your dust

This paragraph is a translation, and there are many versions; however, the key to our exposition is the idea that for an artificial being to be a man, the golem must possess language. The principal method of human communication, consisting of words used in a structured and conventional way and conveyed by speech, writing, or gesture, is a sign of intelligence. The term “intelligent computers” refers to the question “Can computers think” and has ramifications for robotics [24]. There are multiple definitions of AI all around four different axes: (1) thinking-humanly, (2) thinking-rationally, (3) acting-humanly, and (4) acting-rationally [25]. Regarding standard definitions (dictionary), intelligence is the ability to acquire and apply knowledge and skills, the ability to learn or understand or deal with new or trying situations, the skilled use of reason, and the ability to apply knowledge to manipulate one’s environment or to think in abstract terms as measured by objective criteria (such as tests). In simple terms, the act of understanding. This last idea developed within the Thomistic tradition was initially expressed in the writings chiefly by St. Thomas Aquinas [2] and related to the psyche (soul) to acquire knowledge through the internal and external senses² using reasoning. In other words, humankind’s intelligence is:

Human intelligence is the ability to read our experiences and bring to light what is intelligible within them.

The definition is profound in that we are not required to ask questions about the universe’s intelligibility, but we are only required to recognize/understand the world. In other words, something that can be understood by the intellect, not by the external senses. Knowledge is more than sensory experience; but it begins in the senses. If we did not have the sensory experience of the world around us, our minds would be empty.³ In this way, understanding goes beyond perception, like when sight observes the world and our soul (psyche) moves our (internal senses) emotions, i.e., watching our baby fires deep feelings of love. Thus, the term experience goes beyond empirical knowledge and touches the reality we undergo by reasoning and will; this last one is the power that inclines us to what is apprehended as good or fitting.

Figure 2 provides examples of images requiring cognitive abilities to correctly identify the right emotion despite the similarity in visual patterns. The images are superposed with the circumplex model of attention that psychologists apply using valence and arousal scores obtained from directly asking a person who observes the images to rate them. The problem of correctly identifying the right emotion becomes harder to solve by computational methods, as illustrated in Figs. 3 and 4. In both collages, we observe the diversity of patterns that preclude us from directly solving the identification task, and it makes us think of Wittgenstein’s beetle [29]. The beetle

² According to St. Thomas Aquinas, internal senses are common sense, imagination, estimative power, and memory; and external senses are smell, sight, touch, taste, and hearing.

³ CV attempts to recreate the external sense of sight that precedes language; hence representation of thought is the goal we strive to reach by formulating a representation helpful to communicating ideas.

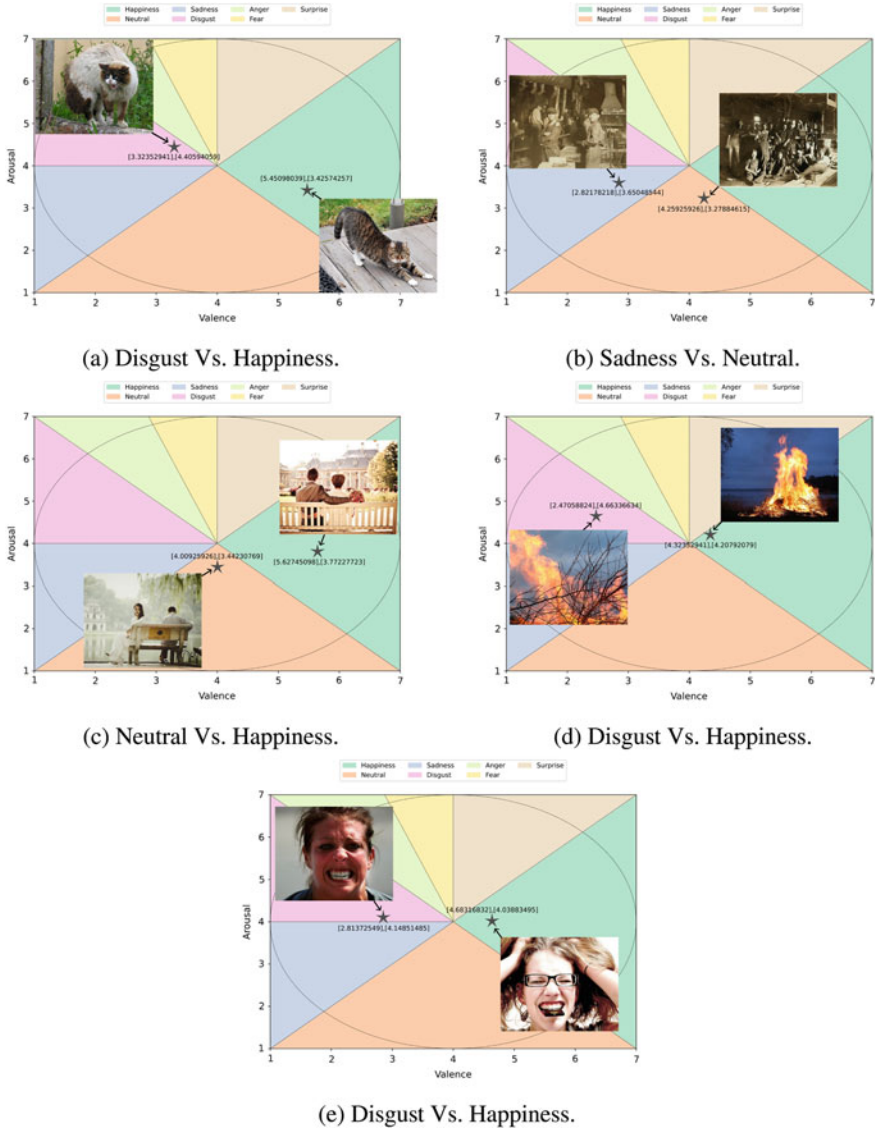


Fig. 2 This figure shows images with common patterns that evoke a contrasting emotional response in humans. The task for the machine is to create a model that correctly elucidates the emotional response



Fig. 3 The idea of happiness is portrayed in OASIS with the above representative set of images. The images have no common set of patterns and we express the associate emotion through metaphors

in the box is an analogy in which everyone has a box that only the owner can see, and no one can see into anyone else’s box. Each person describes what he or she sees in the box as a “beetle.” The paradox is that whatever represents the beetle cannot have a part in the language game since the thing in the box could be changing all the time, like our emotions, or there might be different things in everyone’s box, or perhaps nothing at all in some of the boxes.

4.3 Abductive Reasoning in Brain Programming

This section explains how a new strategy was devised by merging two proposals (Fig. 1d and e) into the flow chart depicted in Fig. 5. Note that we follow a balanced strategy between deductive and inductive reasoning like in previous work about BP. The last work depicted in Fig. 1e successfully provides a methodology capable of separating the foreground from the background on datasets conceived to test salient object detection systems. This knowledge provides a symbolic representation that encapsulates the best possible explanation for what we observe in an image. This



Fig. 4 The idea of sadness is portrayed in OASIS with the above representative set of images. We express the associate characteristics in metaphysical terms since it seems implausible to reach it without valid inferences and using only empirical data

program (rule) derives directly from observations. Abductive reasoning was applied when combining a segmentation step with the saliency map to produce the salient object. The idea is to incorporate the output of such a process (salient object detection) into the AVC model. An intuitive explanation of the abductive reasoning mechanism is as follows: an abductive hypothesis explains a phenomenon by specifying enabling conditions (as a special case) for it. If we want to explain, for example, that the light appears in a bulb when we turn a switch on, an inductive explanation resides in the experience of happening hundreds of times in the past, whereas an abductive explanation bases the analysis on terms of the electric current flowing into the bulb filament [7]. In our example of image classification, the inductive hypothesis appears when we attempt to classify the input through the AVC model since we use the whole image for the computation. However, when we incorporate the salient object detection, we can supply an explanation since we have separated the background from the foreground; thus we can focus more precisely on the part of the image that the system constrains in the computation. This analogy is different from generalization as well as transfer learning. Generalization looks for the ability to achieve good performance under the input of new information, and transfer learning attempts to

Abductive Learning

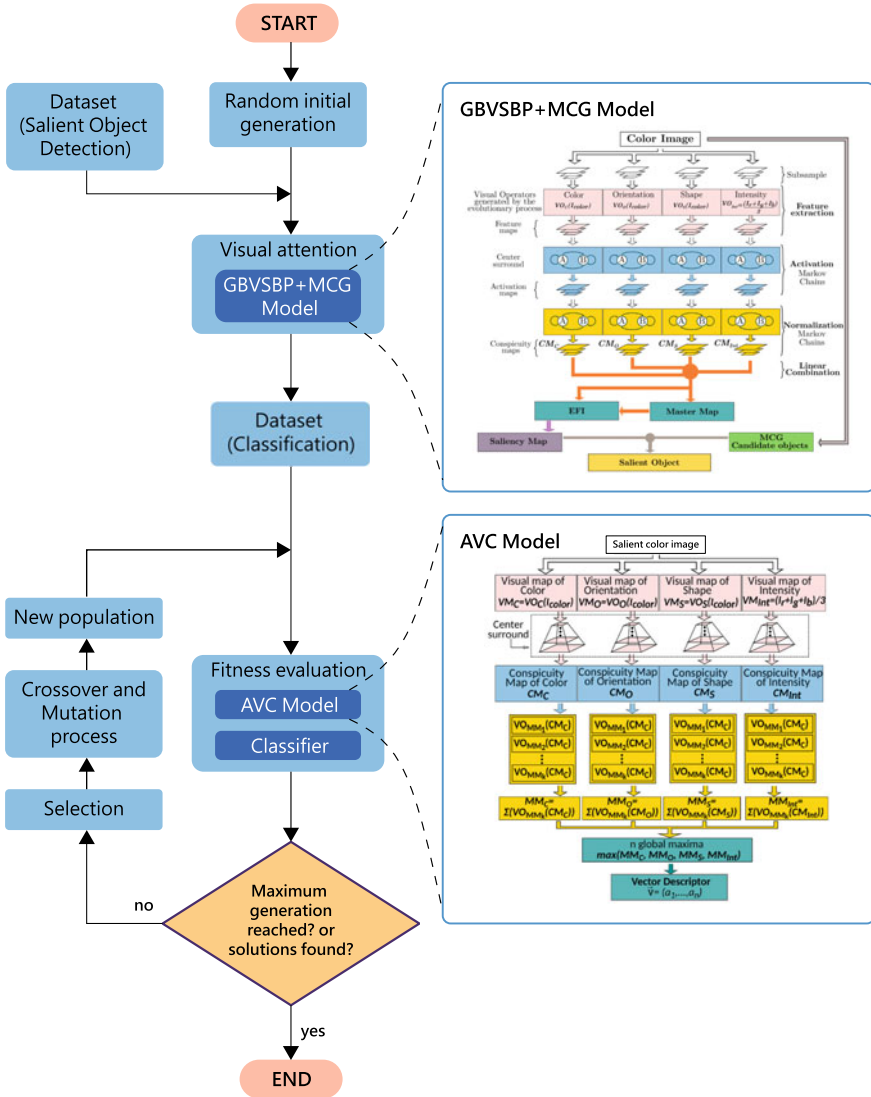


Fig. 5 Flowchart of the abductive brain programming strategy

modify a current model to adapt it to a new problem.⁴ Abduction is an inference to the best explanation; epistemologically, we explain the origin of a new hypothesis by abduction. In other words, an inference based on experience toward the specification of a particular goal while taking chances and making the best of ignorance [1]. Here, knowledge encapsulated in a program/rule for a visual task is applied/connected to another process, similar to the idea of modular brain regions to achieve a specific case. This analysis can take us into an era of resilient system design [5].

5 Results

We propose to attempt to solve with GP and other methodologies (inferential knowledge) the visual Turing test proposed in [19]. The test consists of an image database containing pictures collected by psychologists and normative answers made through a carefully designed process to define a ground truth. Preliminary results show the inability to obtain satisfactory results after probing 40 combinations of five different CNNs (convolutional neural networks) and eight optimizers. The study reflects the problem of current ML methodologies. Indeed, the system memorized all images during the learning stage, see Fig. 6. However, in the testing stage, the score reveals a severe problem since predicted valence and arousal values and the corresponding loss across epochs point to a significant difference that reveals CNNs as useless.⁵ The correlation principle of DL works with data patterns, not with thoughts-ideas-concepts.

Figure 6 shows the results (training and testing) of three different methodologies using accuracy and F1 score. The column on the left portrays AVC statistics, in the middle corresponding to the AVC + VA, and on the right the SqueezeNet with AdaMax. Note that the neural network mimics human behavior through the information provided in the training set. However, the result drastically drops off with the testing data. The number of images having puzzling and cognitively demanding tasks is limited compared to the total number of images.

On the other hand, BP manages to puzzle out the training set without dropping its performance in testing. Figure 7 shows the superposition with the Fibonacci sequence, and the result is astonishing. The Golden Ratio ($\phi = 1.618\dots$) is often called the most beautiful number in the Universe. It appears almost everywhere, starting from geometry to the human body itself. Renaissance artists called this “The Divine Proportion,” and the Fibonacci series appears if we divide a term greater than two by a term preceding it. The Fibonacci spiral is a composition guide that creates a perfectly balanced and aesthetically pleasing image in photography. We hypothesize that images in the dataset follow this principle, and when we adopt the program that

⁴ The literature treated abduction, retroduction, transduction, and even few-shot learning as similar concepts. While tempting to do so, the advice is to study them as different concepts.

⁵ CNNs base their computation on correlations, which is why they are invariant to translation and can store a large amount of data.

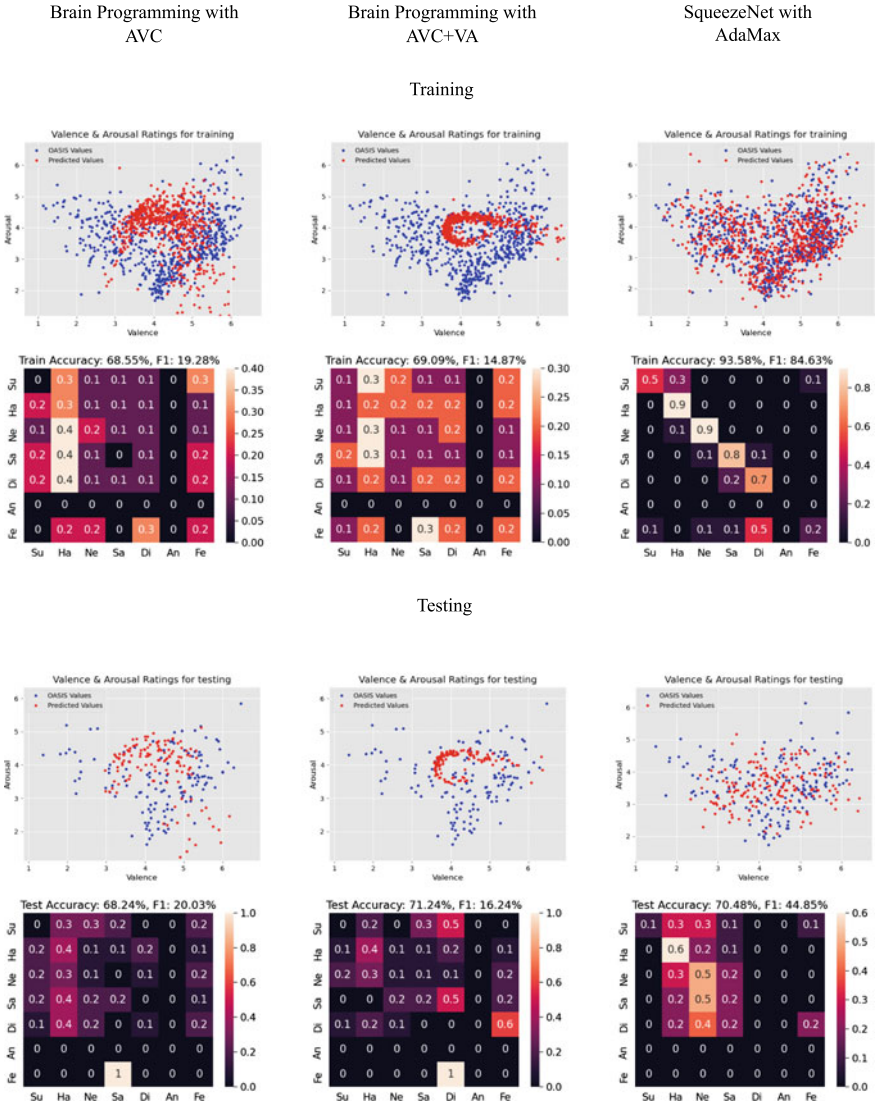


Fig. 6 Machine emotion mapping. Emotion mapping of OASIS images using Brain Programming and a convolutional neural network (Squeezenet with Adamax). Brain Programming’s symbolic learning pattern shows us how the machine was capable of abstracting complex concepts and applying them to unseen images. Meanwhile, the statistical learning of the CNN is an example of how a machine can memorize abstract ideas but fail to use them in unknown environments. BP’s behavior is reminiscent of the question posed to Turing, can machines think? While the machine’s thought process is different from that of humans, this does not mean that the computer is not thinking

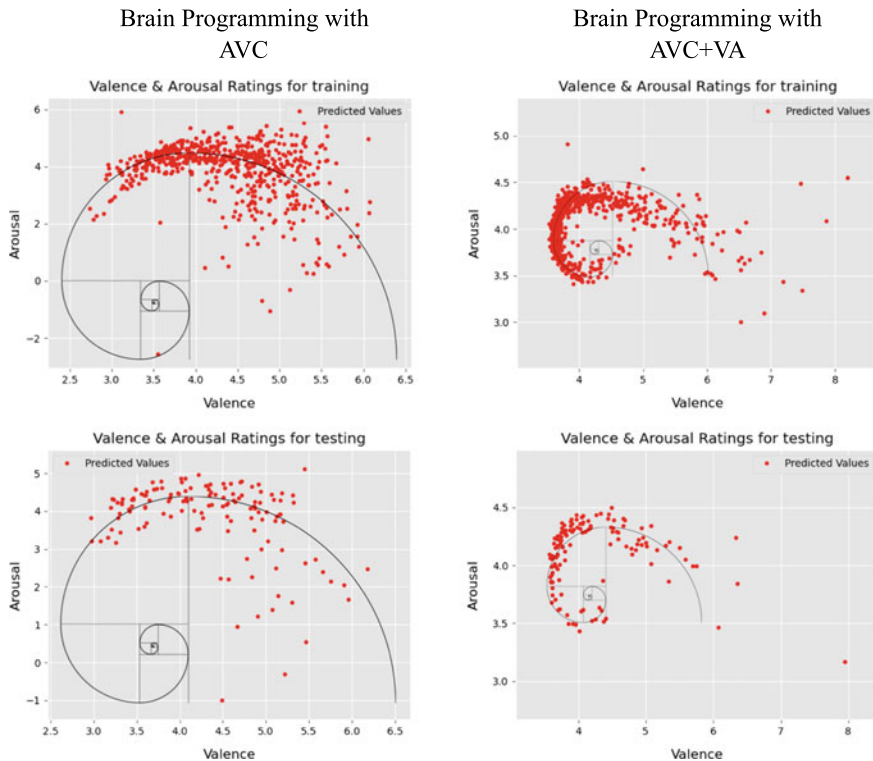


Fig. 7 Machine emotion mapping. A closer look into the pattern formed by Brain Programming when training with the OASIS dataset. If we look closely, the geometry behind the scattered valence and arousal ratings creates a spiral very similar to the one formed by Fibonacci's sequence. This pattern is maintained despite architecture variations and even applied to unseen testing images. The impact of visual attention is also graphically depicted by sharpening the initial shape formed by the AVC

computes the salient object detection, the final result fits this beautiful pattern. This reasoning is a way to understand that the machine/human creates thinking about the OASIS benchmark. The Rule of Thirds and the Golden Ratio materialize through valence and arousal values and can help create/explain a composition/emotion that will draw the eyes/pointer to the essential elements of the photograph. Someone may think that the results did not match the proposed curve; however, we can call to mind some natural phenomena (hurricanes seen from space) that did not perfectly match the Fibonacci sequence but were used to expose the pattern. Nautilus shell is a figure that our outcomes match better for the graceful spiral curve, creating controversy about how accurate it is in following the golden spiral. Understanding that our data is scarce, we do not expect to cover the whole shell, but the vital thing to notice is that independently of what model we are using (AVC or AVC + VA), the output is the same during training and testing.

6 Conclusions

This chapter deals with inferential knowledge and how it is possible to adapt/extend the definitions from social sciences to computer science to construct new knowledge by mimicking thinking in the machine. After an extensive analysis of myths and open issues about GP, ML, CV, and EC, we introduce our approach to synthesizing programs using inferential knowledge and, more specifically, abductive reasoning. BP is a methodology that incorporates domain knowledge at a high level through the idea of templates and at a lower level by selecting the best set of functions and terminals. The resulting symbolic programs provide consistent outcomes for a problem requiring non-contingent representations from the content's viewpoint. The lack of adequacy to a high range of image variation for a single concept is why CNNs fail to model OASIS pictures. In the future, we would like to explore new representations regarding other dimensions, like illumination, that has received little attention from CV and neuroscience communities.

References

1. Andrade, E.: Natural abduction: the bridge between individuals choices and the production of evolutionary innovations. *Signs* **5**, 112–146 (2011)
2. Aquinas, T.: *Summa Theologica* I q 79 a 10. Gilby, T. (ed.) OP, vol. 60. Blackfriars, Cambridge (1966)
3. Choi, C.Q.: 7 Revealing Ways AIs Fail. *IEEE Spectrum* October, pp. 42–47 (2021). <https://spectrum.ieee.org/ai-failures>
4. Clemente, E., Olague, G., Dozal, L., Mancilla, M.: Object Recognition with an Optimized Ventral Stream Model using Genetic Programming. *Applications of Evolutionary Computation*, LNCS, vol. 7248, pp. 315–325 (2012) . https://link.springer.com/chapter/10.1007/978-3-642-29178-4_32
5. Cotta, C., Olague, G.: Resilient Bioinspired Algorithms: A Computer System Design Perspective. *Applications of Evolutionary Computation*, LNCS, vol. 13224, pp. 619–631 (2022). https://doi.org/10.1007/978-3-031-02462-7_39
6. Dozal, L., Olague, G., Clemente, E., Hernández, D.E.: Brain programming for the evolution of an artificial Dorzal stream. *Cognit. Comput.* **6**, 528–557 (2014). <https://doi.org/10.1007/s12559-014-9251-6>
7. Flach, P.A., Kakas, A.C. (eds.): *Abduction and Induction: Essays on their Relation and Integration*. Applied Logic Series. Springer-Science+Business Media, B.V. (2000)
8. Fusing, H., Chen, C., Hsieh, Y.C., Farrell, P.: Lewis Carroll's doublets net of English words: network heterogeneity in a complex system. *PLoS One* **9**(12), e114177 (2014). <https://doi.org/10.1371/journal.pone.0114177>
9. Hernández, D.E., Clemente, E., Olague, G., Briseño, J.L.: Evolutionary multi-objective visual cortex for object classification in natural images. *J. Comput. Sci.* **17**(Part 1), 216–233 (2016). <https://doi.org/10.1016/j.jocs.2015.10.011>
10. Hofman, M.A.: Chapter 18 - design principles of the human brain: an evolutionary perspective. *Progr. Brain Res.* **195**, 373–390. <https://doi.org/10.1016/B978-0-444-53860-4.00018-0>
11. Ibarra-Vazquez, G., Olague, G., Chan-Ley, M., Puente, C., Soubervielle-Montalvo, C.: Brain programming is immune to adversarial attacks: towards accurate and robust image classification using symbolic learning. *Swarm Evolut. Comput.* **71**, 101059, 17 pages (2022). <https://doi.org/10.1016/j.swevo.2022.101059>

12. Koza, J.R.: *Genetic Programming: On the Programming of Computer by Means of Natural Selection*. MIT Press, Cambridge (1992)
13. Koza, J.R.: Human-competitive results produced by genetic programming. *Genet. Program. Evol. Mach.* **11**, 251–284 (2010). <https://doi.org/10.1007/s10710-010-9112-3>
14. Luger, G.F.: *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*. Addison-Wesley (2002)
15. Maynard-Smith, J.: Natural selection and the concept of a protein space. *Nature* **225**, 563–564 (1970). <https://www.nature.com/articles/225563a0.pdf>
16. Mitchell, T.M.: *Machine Learning*. McGraw-Hill (1997)
17. Olague, G.: *Evolutionary Computer Vision: The First Footprints*. Springer (2016)
18. Olague, G., Clemente, E., Hernández, D.E., Barrera, A., Chan-Ley, M., Bakshi, S.: Artificial visual cortex and random search for object recognition. *IEEE Access* **7**, 54054–54072 (2019). <https://doi.org/10.1109/ACCESS.2019.2912792>
19. Olague, G., Olague, M., Jacobo-Lopez, A.R., Ibarra-Vázquez, G.: Less is more: pursuing the visual turing test with the Kuleshov effect. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1553–1561 (2021). <https://doi.org/10.1109/CVPRW53098.2021.00171>
20. Olague, G., Menendez-Clavijo, J.A., Olague, M., Ocampo, A., Ibarra-Vázquez, G., Ochoa, R., Pineda, R.: Automated design of salient object detection algorithms with brain programming 35 pages (2022). [arXiv:2204.03722v1](https://arxiv.org/abs/2204.03722v1). <https://doi.org/10.48550/arXiv.2204.03722>
21. O’Neill, M., Vanneschi, L., Gustafson, S., Banzhaf, W.: Open issues in genetic programming. *Genet. Program. Evol. Mach.* **11**, 339–363 (2010). <https://doi.org/10.1007/s10710-010-9113-2>
22. O’Neill, M., Spector, L.: Automatic programming: the open issue? *Genet. Program. Evol. Mach.* **21**, 251–262 (2020). <https://doi.org/10.1007/s10710-019-09364-2>
23. Rich, C., Waters, R.C.: Automatic programming: myths and prospects. *Computer* **21**(8), 40–51 (1988). <https://doi.org/10.1109/2.75>
24. Rosenfeld, A.: Religion and the robot. *Tradit: J. Orthodox Jewish Thought* **8**(3), 15–26 (1966). <http://www.jstor.org/stable/23256081>
25. Rusell, S., Norvig, P.: *Artificial Intelligence a Modern Approach*, 3rd edn. Prentice Hall (2010)
26. Salisbury, F.B.: Natural selection and the complexity of the gene. *Nature* **224**, 342–343 (1969). <https://www.nature.com/articles/224342a0.pdf>
27. Samuel, A.L.: Some studies in machine learning using the game of checkers. *IBM J. Res. Develop.* **3**(3), 210–229 (1959). <https://doi.org/10.1147/rd.33.0210>
28. Turing, A.M.: Computing machinery and intelligence. *Mind* **LIX**(236), 433–460 (1950). <https://doi.org/10.1093/mind/LIX.236.433>
29. Wittgenstein, L.: *Tractatus Logico-Philosophicus*, first English-language edition, Harcourt, Brace & Company, Inc (1922)