The Image Clustering Technique used to Find Density of Data Points in An Image using Hill Climbing Algorithm Technique

Rajshekhar Ghogge[1]
[1]Department of ISE
[1]Dr. AIT, Bangalore, Karnataka, India

Abstract—Images are considered as one of the most important medium of conveying information. Understanding images and extracting the information from them such that the information can be used for other tasks is an important aspect of Machine learning. In this paper various clustering techniques along with some clustering algorithms are described. Further Hill climbing algorithm, its limitations and a new approach of clustering called as M-step clustering that may overcomes these limitations of k-means is included. Image retrieval is the basic requirement task in the present scenario. Content Based Image Retrievalis the popular image retrieval system by which thetarget image to be retrieved based on the useful features of the given image. In other end, image mining is the arising concept which can be used to extract potential information from the general collection of images. Target or close Images can be retrieved in a little fast if it is clustered in a right manner.

Keywords: Image Clustering, Hill climbing, K-means, M-step, Density of Images

I. INTRODUCTION

Evolutionary algorithms are the ones that follow the Darwin concept of "survival of the fittest" mainly used for optimization problems for more than decade. The performance of super computers can be enhanced by parallelizing compilers to check the problem of data usage which can be reduced to characterization of a Diophantine equation [Zhiyu 1989].Computable economics [Velu 2004] uses decision problems like Diophantine equations to propose a change in the market equilibrium conditions instead of the conventional parameters. Evolutionary algorithms are heuristic search algorithms which do not always guarantee that to provide the exact optimal solutions, but they will definitely find better[1]. optimal solution within less amount of time of which some of Genetic algorithms are used[1]. A chromosome contains a group of numbers that completely specifies a candidate solution during the optimization process. The individuals with higher fitness values will survive and will be selected to produce a better generation, while the individuals with lower fitness values will be eliminated. The paper intends to find high quality saliency maps of the same size and resolution as the input image and their use in segmenting whole objects[2].

The method is effective on a wide range of images including those of paintings, video frames, and images containing noise. All methods use some means of determining local contrast of image regions with their surroundings using one or more of the features of color, intensity, and orientation. Our method for finding salient regions uses a contrast determination filter that operates at various scales to generate saliency maps containing saliency values" per pixel.Saliency maps are created at different scales use saliency calculation method described below. Thereafter all these maps are added pixel wise to get the final saliency maps. The input image is then over-segmented and the segments whose average saliency exceeds the threshold value are selected[3].

The crossover operator is complicated process the exchanges the substrings between two chromosomes depends on the encoding schemes. In order to increase the performance of genetic algorithms chromosomes must be applied. The performance of genetic algorithms depends on the balancing between the exploitation and exploration techniques.Exploitation means to use the already available knowledge to find out the better solution and Exploration is to investigate new and unknown area in search space[4]. The power of genetic algorithms comes from their ability to combine both exploration and exploitation in an optimal.

II. RELATED WORK

The performance of super computers can be enhanced by parallelizing compilers to check the problem of data usage which can be reduced to characterization of a Diophantine equation [Zhiyu 1989]. Computable economics [Velu 2004] uses decision problems like Diophantine equations to propose a change in the market equilibrium conditions instead of the conventional parameters. Integer factorization [Knuth 1997] uses Diophantine equations in the process of breaking down a composite number into smaller non-trivial divisors. Diophantine equations are also used in other areas like algebraic curves [Ponnen 2000], projective curves [Brown & Myres 2002] [Stroeker & Tzanakis 1994] and theoretical computer science [Ibarra 2004][Guarari 1982]. These application areas make Diophantine equations an important domain not just in the realm of Mathematics but in other fields too[5].

Heuristic search algorithms have exponential time and space complexities as they store complete information of the path including the explored intermediate nodes. Hence many applications involving heuristic search techniquesto find optimal solutions tend to be expensive. Despite of these, the researchers have strived to find optimal solution in best possible time. In this paper we have considered major algorithms which are applied to find the shortest path: hill – climbing, steepest –ascent, best first and A* [1,2,4][6].

Hill climbing algorithms expand the most promising descendant of the most recently expanded node until they encounter the solution. Steepest – ascent hill climbing differs from hill climbing algorithm only the way in which the next node is selected. In this method it selects best successor node for expansion, unlike the first successor node for expansion, as done in hill climbing. Though this method tries to choose best possible path , but this method , like hill climbing method may fail to find a solution by reaching to a node from were no improvements can be done [5,8]. The methods that can be optimized and cont Rolled access hill climb algorithm and which can be further

analyzed the various techniques that be modified and organized by various method of scheduling the hybrid hill climb algorithm and that can be overtaken by specific rules and condition.

A. Local Search and Learning:

Local search methods use local knowledge to improve a solution's chances to propagate its characteristics into the next generations. Due to the similarities in the role of the local search within the genetic search and the role of learning within the evolution process, the local search is usually viewed as a learning process. Lamarckian evolution and Baldwin effect: One of the important issues of hybrid genetic algorithms is how the information gained during local search is used by the global algorithm [7].

Either the Lamarckian or the Baldwin approach can be used. In the Lamarckian approach the traits acquired during the learning process are passed from parents to their offspring. This means that both the genetic structure of an individual and its associated fitness value are modified to reflect the changes in phenotype structure as a result of performing local search. The Baldwin Effect is somewhat Lamarckian in its results but using different mechanisms. In the Baldwin approach the learning process can help the individual to adapt to its environment and as a result to survive [8].

B. Balance between Global and Local Search:

The hybrid algorithm should strike a balance between exploration and exploitation, in order to be able to solve global optimization problems. According to the hybrid theory solving an optimization problem and reaching a solution of desired quality can be attained in one of two ways. Either the global search method alone reaches the solution or the global search method guides the search to the basin of attraction from where the local search method can continue to lead to the desired solution[9].

In the genetic-local hybrid, the main role of the genetic algorithm is to explore the search space in order to either isolate the most promising regions of the search space, or, to hit the global optimum. However, the main role of the local search method is to exploit the information gathered by the global genetic algorithm[10]. The division of the hybrid's time between the two methods influences the efficiency and the effectiveness of the search process. The optimal division of the algorithm's time is an important issue that is faced the method.

III. PROPOSED METHOD

A. Mimetic Algorithm Approach:

Incorporating problem specific information in a genetic algorithm at any level of genetic operation form a hybrid genetic algorithm. The technique of hybridization of knowledge and global genetic algorithm is memetic algorithm (MA). MA is motivated by Dawkins notation of a meme. A meme is a unit of information that reproduces itself as people exchange ideas. MA binds the functionality of GA with several heuristic's search techniques like hill climbing, simulated annealing, Tabu search etc[11]. A number of issues should be carefully addressed when an effective hybrid genetic algorithm is constructed .Two popular ways of hybridization depends on the concepts of "Baldwin effect" and "Lamarckism". According to Baldwinian search strategy, the local optimization can interacts and allow the local search to change the fitness of individual but genotype itself remain unchanged. The disadvantage of Baldwinism is that it is slow [12].

According to Lamarckism, the characteristics acquired by individual during its lifetime may become heritable traits. According to this approach both the fitness and genotype of individuals are changed during local optimization phase. Most of the MA is based on Lamarckism approach of hybridization. The proposed memetic algorithm incorporates hill climbing local search after selection process in order to increase exploitation. In the proposed approach, members selected using roulette wheel selection has been used as initial point to carry out hill climbing search. In this approach, each individual is improved using hill climbing before passing to reproduction phase[13].

1) Algorithm 1: Proposed Hybrid Genetic Algorithm

Procedure MA(fitfxn, psize, Pc, Pm)
fitfxn – fitness function to evaluate chromosome
psize – size of population in each generation
Pc – crossover probability
Pm – mutation probability
mxgen –maximum number of generations encode solution space
Initialize population gen=1
while (gen <=
mxgen) evaluate (min(fitfxn)) for i = 1 to psize
mate1, mate2=select(population)
// apply local search to each selected individual optmate1=hill climbing (mate1) optmate2=hill climbing (mate2)
If (rnd(0,1)<=Pc)
child = crossover(optmate1, optmate2) end If
If (rnd(0,1)<=Pm)
mchild = mutation(child) End If
End for
Add offspring to new generation gen=gen+1
End while
return best chromosomes

B. Hill Climbing Local Search:

Hill climbing is an optimization algorithm for single objective function. In hill climbing algorithm a loop is performed in which the currently known best individual produce one offspring. If the fitness of new individual is better than parent it replaces it, else stop the loop[14].

1) Algorithm 2: Hill Climbing Algorithm

Procedure Hill climbing (parent) //parent – currently known best solution
While (termination criteria is not specified) do New_solution <– neighbors (parent)
If (New_Solution is better than parent) Parent = New_solution
End If
End While
return best solution

The analysis results showed that memetic algorithm converges faster than genetic algorithm and produces more optimal result. A comparative analysis of memetic algorithm based on hill climbing search and

genetic algorithm has been performed for the cryptanalysis on simplified data encryption standard problem by Poonam Garg[15]. She concluded that memetic algorithm is superior for finding number of keys than genetic algorithms. This method that can be optimized.

IV. METHODOLOGY

Procedure for memetic algorithm is same as simple genetic algorithm except that a local search method is implemented in one of the operator (crossover, selection, replacement) to exploit the search space. Applying Hill climbing in replacement operator work efficiently to find the optimal solution.

Simple GA represents an intelligent exploration, having a random search confined within a defined search space for solving a problem optimally. Simple GA starts with random initialization of population. After this fitness function is used to calculate the fitness of each individual and then reproduction is applied. In order to incorporate the offspring into original population replacement is used. Various replacement schemes are used for maintaining the useful diversity of population.

Elitist replacement schemes improve the performance of genetic algorithm. Using different replacement and selection schemes in steady state, genetics converge quickly and have a useful diversity. Diversity helps in finding the optimal solution. The time needed to reach the global optimum can be reduced if local search methods and local knowledge are used to accelerate locating the most promising region in addition to locating the global optimum starting within basin of attraction. Meta heuristic search mechanism in the memetic algorithm offers the speed and quality of convergence. Reducing the population size can lead to an increase in the algorithm convergence speed [16].

In memetic algorithm, in spite of using the basic generational update, hill climbing helps in finding the better individuals for replacement. These improvements accumulate over all the generations, resulting in a larger improvement in the total performance. Genetic algorithm and local search have complementary properties, which helps in optimization of objective function with fast convergence. When evaluating the fitness of each individual, GA use the results of HC working with an initial guess corresponding to this individual, there are thus as many HC running in parallel as individuals in the population. During reproduction and genetic transformation (crossover, mutation) for the production of the individuals of the next generation, GA work on the new solution [17].

Though Diophantine equations have a great historical background and have been used in many areas, there does not exist a general method to find solutions of such equations [Davis 1992] [Matiyasevich 1993]. Then, finding numerical solutions to such equations is the only next way out. This is a tough task as the computing complexity involved in such a process is quite high. In this regard, applying artificial intelligence techniques, which are known for maneuvering huge search space, is significant. Literature talks about few attempts to find numerical solutions of Diophantine equations using hard computing and soft computing techniques of Artificial Intelligence. Abraham and Sanglikar [Abraham and Sanglikar 2001] used basic genetic operators like mutation, inversion and crossover [Michalewich 1992] to find numeric solutions of some elementary equations. They [Abraham and Sanglikar 2007 a] later used a procedure called 'host parasite co-evolution' [Hills 1992] [Paredis 1996][Wiegand 2003] in a typical genetic algorithm to find numerical solutions. They also proposed [Abraham and Sanglikar 2007 b] a unique evolutionary and co-evolutionary [Rosin and Belew 1997] computing method to find numerical solutions of such equations [18].

This paper proposes hill climbing as a hard computing artificial intelligence technique to find numerical solutions of Diophantine equations. Hill Climbing is a local search [Russel & Norwig 2003] technique. It starts with an initial solution and steadily and gradually generates neighboring successor solutions. If the neighboring state is better than the current state, we make the neighboring state the current state [19].

The whole process can be taken as an optimization process [Lugar 2006]. There are different variants of hill climbing. They are simple hill climbing, steepest hill climbing, stochastic hill climbing and random restart hill climbing. The paper uses steepest ascent version of the hill climbing to find numerical solution of Diophantine equations. In steepest hill climbing all successor nodes are probed and compared for its relevance and then the best amongst them is taken as the successor node. This results in having an exhaustive local search and identification of the best possible successor of a given node at any instant of time. The hill climbing algorithm which can be optimized [20].

V. FLOW CHART

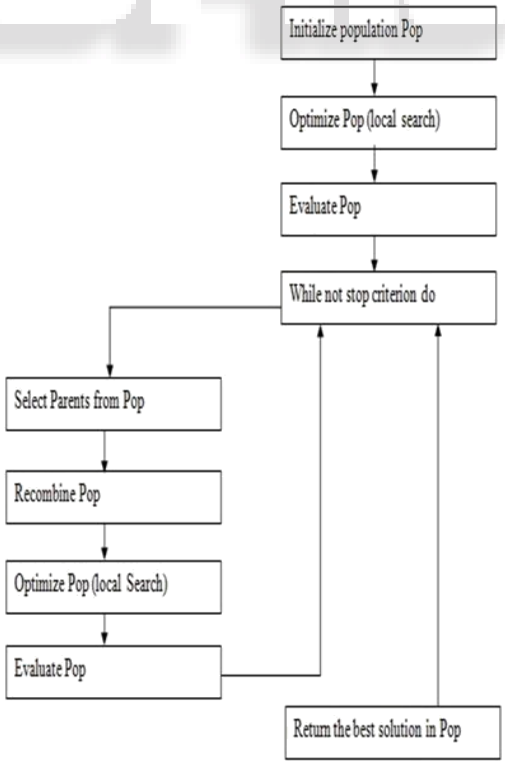

Fig. 1: Two Possible Ways to Combining Local Search with SGA

VI. RESULTS AND DISCUSSION

A. *Advantages of the System:*

- The high climb algorithm it can be used as artificial intelligence method.
- The high climb algorithm which can be measured as content of producing high optimization.
- The general method image processing and image segmentation can done by using this algorithm.

B. *Disadvantages of the System:*

- The main disadvantage of the high climb algorithm system it is complex process.
- Another disadvantage of the system is that the various steps are required to achieve a single process.
- High climb algorithm which involves the complex code that normal user cannot understand.

VII. CONCLUSION

We can conclude Image segmentation is a very important part of image processing. Detection of salient image regions is useful for applications like image segmentation, adaptive compression, and region-based image retrieval. This problem can be tackled by mapping the pixels into various feature spaces, which are subjected to various grouping algorithms It present a novel method to determine salient regions in images using low-level features of luminance and color. The method is fast, easy to implement and generates high quality saliency maps of the same size and resolution as the input image.

The artificial intelligence method of which it can be concluded that image segmentation and image processing which can be generalized to extended procedure in future the simplex code can be developed so that simple user can understand method of accessing code very specific method and which combindly produces general procedures to use high climb algorithm where data integrity and data segmentation can be possible general method which can be explained in understanding manner.

REFERENCES

[1] BlaiBonet and Eric A.Hansen,(2010)"Heuristic Search for Planning under Uncertainty", Chapter in Heuristics, Probability and Causality: A Tribute to Judea Pearl College Publications. pp 3-22
[2] Eric A Hansen, Rong Zhou, (2007) "Anytime Heuristic Search", Journal of Artificial Intelligence Research 28, pp 267-297
[3] G.Cornuejols and G L Nemauser, (1978) "Tight bounds for christofides" travelling salesman heuristic* Short Communication Mathematical Programming, Vol. 14, Issue 1, pp 116-121
[4] Anne L. Gardner, (Sept 1980) "Search: An Overview", AI magazine, Vol. 2, Number 1
[5] R. Korf, (1990) "Real time heuristic search", Artificial Intelligence ACM Digital Library, Vol. 42,pp189-211
[6] RinaDechter and Judia Pearl, (July 1985) "Generalized Best-First Search Strategies and the Optimality of A*.", Journal of the Association for Computing Machinery, Vol. 32, No. 3, pp 505-536
[7] L. Fu, D. Sun and L. R. Rilett, (2006) "Heuristic shortest path algorithms for transportation applications: State of the art", Elsevier Computer and Operations research 33, pp 3324-3343
[8] Girish P. Potdar andDr.R.C.Thool, (2013) "An Alternate way of implementing Heuristic Searching Technique" International Journal of Research in Computer and Communication Technology, Vol. 2, No 9, pp-793-795
[9] Hen-Yong Pang, Alicia Tang Y.C., (2006) "A Route Advisory System (RAS) For Travelling Salesman Problem", Journal of Applied Sciences Research 2(1), pp 34-38
[10] C.H. Peng, J.S. Wangand R.C.T. Lee, (1994)"Recognizing Shortest Path Trees in Linear Time", Information Processing Letters, Vol. 57, pp 77-85
[11] J. Holland, Adaptation in natural and artificial systems, University of Michigan Press, Ann Arbor, 1975.
[12] D.E. Goldberg, Genetic algorithms in search, optimisation, and machine learning, Addison Wesley Longman, Inc., ISBN 0-201-15767-5, 1989.
[13] C. Ryan, Automatic re-engineering of software using genetic programming, Genetic Programming Series. Kluwer Academic Publishers, ISBN 0-7923-8653-1, 2000.
[14] W. E. Hart, "Adaptive global optimization with local search, Doctoral diss.", San Diego, University of California, 1994.
[15] D. E. Goldberg and P. Segrest, "Finite Markov chain analysis of genetic algorithms", Proceedings of 2nd International Conf. on Genetic Algorithms, Lawrence Erlbaum Associates, 1987, pp 1-8.
[16] L .Booker, Improving search in genetic algorithm, genetic algorithm and simulated annealing, Pitman,1987, pp 61-73, vol 5.
[17] Rakesh kumar and Jyotishree, "Novel knowledge based tabu crossover in genetic algorithms", International Journal of Advanced research in Computer science and software Engineering, vol 2, No. 8, pp 78-82, Aug 2012.
[18] H. A. Sansi, A. Zubair and R. O. Oladele, "Comparative assessment of genetic and memetic algorithms", Journal of Emerging Trends in Computing and Information Science, vol 2, No. 10, pp 498-508, Oct 2011.
[19] Poonam Garg, " A comparison between memetic algorithm and genetic algorithm for the cryptanalysis of simplified data encryption standard algorithm", International Journal of Network Security and its Applications,vol 1, No. 1, pp 34-42, April 2009.
[20] Antariksha Bhaduri, "A mobile robot path planning using genetic artificial immune network algorithm", Proceedings of World Congress on Nature and biologically Inspired Computing, NaBIC, IEEE, 2009, pp 1536-1539.

