Using Shared Multi-Terminal Binary Decision Diagrams for Image Representation

Miloš M. Radmanović[1]

Abstract – **Compactness criteria for 2D image representations are important because of their increasing role in many computer applications. This paper discusses the use of shared multi-terminal binary decision diagrams (SMBDDs) to represent bitmapped image. In order to investigate and compare performance of the SMTBDD image representation, I have developed a specialized tool for building an SMTBDD from bitmapped image. I have demonstrated that SMTBDDs are an efficient representation for every special-case image.I do not purport to "prove" in any real sense that SMTBDDs are a superior representation of general images. This paper is not the end, but rather the beginning of my research.**

Keywords – **Boolean functions, shared multi-terminal binary decision diagram, image representation, software tool**

I. Introduction

Binary Decision Diagrams (BDDs) are a data structure that has been used for years to provide an efficient representation of Boolean functions. BDDs were introduced by Akers in 1959 [1]. In the early 1980's, Bryant [2] demonstrated how a BDD could be modified to become a canonical representation of a Boolean function. In 1990's, Coudert, and Madre [3] demonstrated that BDD could represent sets of finite-state machine states efficiently.

In [4], it is observed that for a BDD is generally thought to take only terminals 0 and 1.It is shown that a BDD can have integer terminals (Multi-terminal BDD - MTBDD). In the late 1990's, there are many explorations of the relationship between BDDs and matrices. Fujita, McGeer, and Yang [5] have demonstrated that MTBDDs are the space-optimal representation of dense, sparse,and permutation matrices. In [6], it is proposed a new compression scheme to reduce the huge size of the inverted files in a large information retrieval system without loosing the querying efficiency. The basic idea is to transform the inverted list into a logic function, to represent that function in the form of BDD, and then store the BDD directly in the inverted file.

In [7], it is shown how images can be compressed using Ordered Binary Decision Diagram (OBDD). The mechanism of sharing identical sub-OBDDs representing subimages is also useful when compressing related image sequences such as movies.However, the compression ratios obtained with this approach are low. In [8], it is presented a coding algorithm for OBDDs which provided better compression ratios. It has also been shown that performances of image operations on BDDs depend on the size of the BDDs [9].In [10], it is shown that the geometric transformations of an image represented by a BDD can be expressed using only BDD manipulation process.

In this paper, I discuss the use of Shared MTBDD (SMTBDD) [10] to represent images. I take ideas introduced in [5], [7], and [8], and extend them to SMTBDD. The basic idea of my approach is to represent the image to be coded with a Boolean function, and then simplify and code it efficiently with SMTBDD. In order to investigate and compare performance of SMTBDD image representation, I have developed a specialized tool for building SMTBDD from bitmapped image.

This paper is organized as follows: Section 2 shortly introduces the SMTBDDs. Section 3 describes SMTBDD bitmap image representation. Section 4 describes the software tool for building SMTBDDs from bitmap images and shows experimental analysis of SMTBDD representation for some benchmark bitmap images. I finally give some conclusions in section 5.

II. Shared Multi-Terminal BDD

Binary Decision Diagrams (BDDs) are data structure convenient for representation of discrete functions. Due to that, BDD have become widely used for a variety of CAD applications, including symbolic simulation, verification of combinational logic, and verification of sequential circuits, see for instance [11], [12].

BDDs are derived by the reduction of the corresponding binary decision trees (BDTs). The reduction is performed by sharing the isomorphic subtrees and deleting the redundant information in the BDT using the suitably defined reduction rules [2].

Multi-terminal binary DDs (MTBDDs) are generalization of BDDs. They can represent Boolean functions by the corresponding integer equivalent functions [11]. This technique is useful for various areas in computer science.

Multiple-output switching functions are represented by shared BDDs (SBDDs) or shared MTBDDs (SMTBDDs) having a separate root node for each output.

An example of SMTBDD for the multiple-output function with outputs f_0 and $f_1$is shown in the following figure.The truth vector with integer values of the function f_0 is F_0 = [2, 2, 4, 2, 4, 4, 4, 4]^T and of the function f_1 is F_1=[2, 2, 4, 2, 4, 3, 4, 4]^T.

It is obvious that this SMTBDD is quite smaller then two MTBDDs for the functions f_0 and f_1 in the number of nodes since there are shared values of the vectors of f_0 and f_1.This feature is essential in SMTBDD representations and will be highly exploited in SMTBDD representation of bitmapped image.

[1]Miloš M. Radmanović is with the Faculty of Electronic Engineering, Aleksandra Medvedeva 14, 18000 Niš, Serbia, E-mail: milos.radmanovic@gmail.com

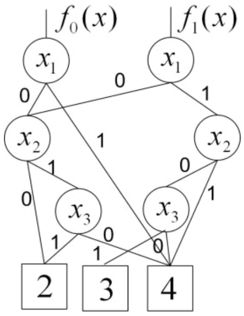

Fig. 1. An example of SMTBDD for the functions f_0 and f_1.

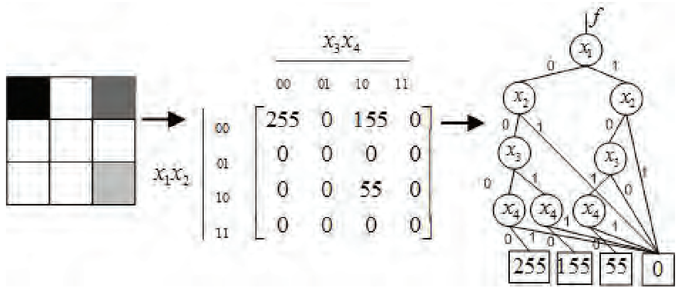

Fig. 3. An example of MTBDD representation for gray-scaled bitmapped image of size 3×3

III. SMTBDD Bitmap Image Representation

Typically,the black and white bitmapped image can be representing as a matrix of size $n\times m$ whose entries correspond to pixels having either value 0 or 1.For gray-scaled bitmapped images, matrix entries correspond to pixel values between 0 and 255. For RGB color bitmapped images, R, G, and B matrices entries correspond to pixel values between 0 and 255.

To transform a bitmapped image to a BDD, it requirescreationof a set of Boolean function variables to each entry of the image matrix and consideration of each pixel asa minterm representation of the Boolean function. Thereafter, for image matrix of size $n\times m$,it is necessary $[\log_2 n]+[\log_2 m]$ variables [9]. Then, mintern representation of the Boolean function can be transform to BDD using BDD construction algorithms proposed by researchers [2], [3], [4]. An example of BDD representation for black and white bitmapped image is shown in the following figure.

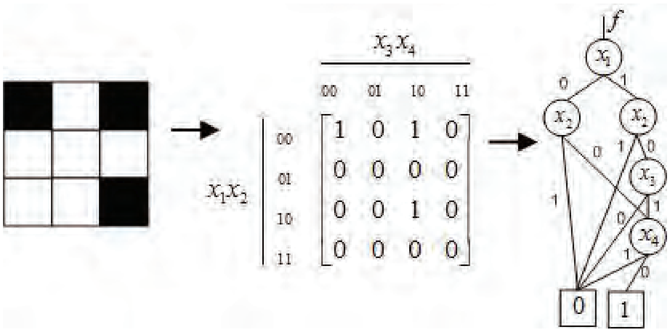

Fig. 2. An example of BDD representation for black and white bitmapped image of size 3×3

For gray-scaled bitmapped images, mintern representation of Boolean functions with integer values can be transform to MTBDD using MTBDD construction algorithms proposed by researchers [5]. An example of MTBDD representation for gray-scaled bitmapped image is shown in the following figure.

For RGB color bitmapped images, I propose mintern representation of multi-output Boolean functions with integer values. This representation can be transform to SMTBDD using SMTBDD construction algorithms proposed by researchers [10]. An example of SMTBDD representation for RGB color bitmapped image is shown in the following figure.

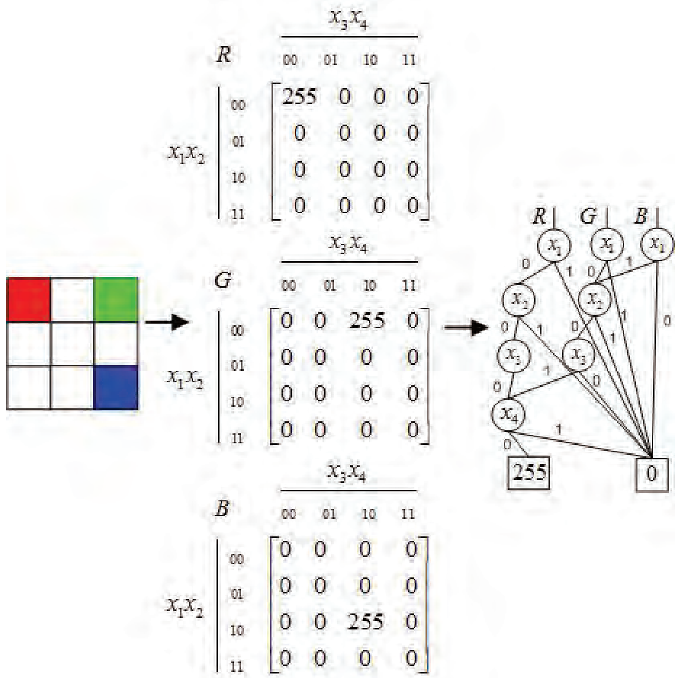

Fig. 4. An example of SMTBDD representation for RGB colorbitmapped image of size 3×3

It is obvious that SMTBDD representation for RGB color bitmapped image from previous example is quite smaller then three MTBDDs for the multi-output function with outputs R, G and Bin the number of nodes since there are shared values betweenR, G and B matrices.

It should be observed that SMTBDD from previous example is compact since there are many constant subvectors of four consecutive 0 in R, G and B matrices.

IV. EXPERIMENTAL RESULTS

In order to investigate and compare performance of SMTBDD image representation, I have developed a specialized tool for building SMTBDD from bitmapped RGB color images.

Specialized software tool is written in MS Visual C++ and use MFC technology [13]. It consists of three basic modules: (1) BDD module for SMTBDD representation of Boolean functions [14], (2) Image transformation module for minterm representation of RGB color bitmapped images, and (3) Interaction module that allow user to interact with software tool.

Below I give a list of RGB color image benchmarksof various types. Benchmarksare based on the collection of famous image benchmarks freely available in the public domains[15], [16]. The benchmarks then have been categorized in four categories: benchmarks of detailed type (Figure 5), benchmarks of screen test type (Figure 6), benchmarks of text type (Figure 7) and, benchmarks of texture type (Figure 8). This allows judging the quality of SMTBDD image representations and gives a better overview of representation performance. Descriptions of benchmarks are given in table 1.

I performed the testing on a PC Pentium IV on 2,66 GHz with 4GB of RAM on MS Windows 7 platform. The memory usage for tool was limited to 2 GB, and space statistics of benchmarks is presented in table 1. All benchmarks are 24-bit RGB bitmapped images.

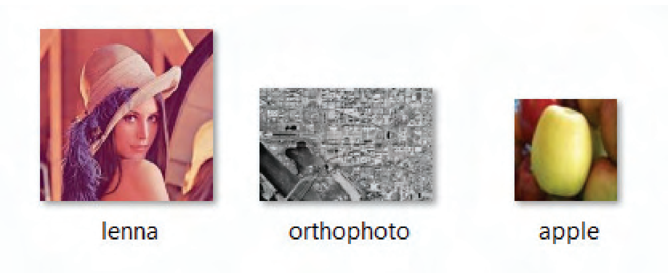

Fig. 5. Benchmarks - RGB color images of detailed type

Fig. 6. Benchmarks - RGB color images of screen test type

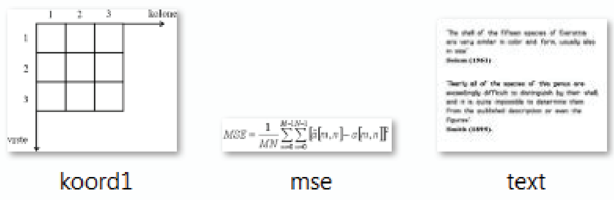

Fig. 7. Benchmarks - RGB color images of text type

Fig. 8. Benchmarks - RGB color images of texture type

TABLE I
SPACE STATISTIC OF SMTBDD REPRESENTATION FOR RGB COLOR BITMAPPED IMAGES

Image	Size [pixels]	Image size [elem.]	SMTBDD size [nodes]	Comp. ratio [%]
lenna	204 x 204	124848	73461	58
orthophoto	512 x 256	393216	86564	22
apple	64 x 64	12288	9731	79
g04	128 x 64	24576	411	1
g08	128 x 64	24576	2516	10
g24	128 x 64	24576	3109	12
koord1	389 x 324	378108	2168	0.5
mse	285 x 54	46170	1611	3
text	111 x 86	28638	4154	14
brain	249 x 203	151641	68143	45
textile	225 x 225	151875	108590	71
texture	259 x 194	150738	82314	55

Table 1 describes SMTBDD space statistics using proposed representation of RGB color bitmapped images. The third column of the table presents the total number of elements in three matrices (R, G and B) of the matrix representation for bitmapped images. The fourth column of the table presents total number of nodes in SMTBDD representation for bitmapped images. Differences in space performances between RGB matrices and SMTBDD image representation are emphasized by percentage values in the last column of the table.

According to percentage values of differences, in most cases for details and texture type images, it is shown that SMTBDD is not compact representation. But, in all cases for screen test and text type images, it is shown that SMTBDD is very efficient representation.

V. CONCLUSION

This paper describedthe use of SMTBDDs for compact representation of RGB color bitmapped images. In order to investigate performance of SMTBDD image representations, I have developed a specialized software tool for building SMTBDD from matrix representation of bitmapped image. I just report the fact that first experimentation with SMTBDD image representation is quite satisfactory. It opens the possibility to further research of SMTBDD representation for some special-case images. I do not purport to “prove” in any real sense that SMTBDDs are a superior representation of general images. This paper is not the end, but rather the beginning of my research.

REFERENCES

[1] S. Ackers, “Binary Decision Diagrams”, IEEE Trans. on Computers, vol. C-27(6), pp. 509-516, 1978.

[2] R. Bryant, "Graph-based Algorithms for Boolean Function Manipulation", IEEE Trans. Computers, vol. C-35, pp. 667-691, 1986.

[3] O. Coudert and J.C. Madre, “A unified framework for the formal verification of sequential circuits,” IEEE Int. Conf. on CAD, 1990.

[4] E. Clarke, M. Fujita, K. McMillan, J. Yang, X. Zhao, “Spectral Transforms for Large Boolean Functions with applications to technology mapping”, Proc. of the 30th Int. Design Automation Conf., 1993.

[5] M. Fujita, P. McGeer, J. Jang, “Multi-Terminal Binary Decision Diagrams: An Efficient Data Structure for Matrix Representation”, Journal of Formal Methods in System Design”, vol. 10, is. 2-3, 1997.

[6] C. Lai, T. Chen, “Compressing Inverted Files in Scalable Information Systems by Binary Decision Diagram Encoding”, ACM/IEEE 2001 Conference, 2001.

[7] M. Starkey, R. E. Bryant,”Using Ordered Binary Decision Diagrams for Compressing Images and Image Sequences” Technical Report, CMU-CS-95-105, CarnegieMellonUniversity, 1995.

[8] P. Mateu-Villarroya, J. Prades-Nebot, “Sequential Logic Compression of Images”, Proc. of IEEE Int. Conf. on Image Processing (ICIP'01), Tessaloniki, Greece, pp. 479-482, 2001.

[9] C. Lursinap, K. Kanchanasut, T. Siriboon, "Basic Binary Decision Diagram Operation for Image Processing" Advances in Computer Science - ASIAN'97, Lecture Notes in Computer Science, 1997, vol. 1345, pp. 368-370, 1997.

[10] L. Robert, G. Malandain, "Fast Binary Image Processing Using Binary Decision Diagrams", Computer Vision and Image Understanding, Elsevier, vol. 72-1, pp. 1-9, 1998.

[11] T. Sasao, M. Fujita, “Representations of Discrete Functions”, Kluwer Academic Publishers, Boston, 1996.

[12] G. Hachtel. F. Somenzi, Logic Synthesis and Verification Algorithms, Kluwer Academic Publishers, 1996.

[13] Microsof Visual Studio 6.0, Microsoft Corporation, http://msdn.microsoft.com/en-us/library/ms950417.aspx

[14] Stgroup CIITLAB Software, http://stgroup.elfak.edu.rs/home/node/4

[15] Computer Vision Test Images, http://www.cs.cmu.edu/~cil/v-images.html

[16] BMP Suite, http://entropymine.com/jason/bmpsuite/