L2/09-273

ISO/IEC JTC 1/SC 2/WG 2
PROPOSAL SUMMARY FORM TO ACCOMPANY SUBMISSIONS
FOR ADDITIONS TO THE REPERTOIRE OF ISO/IEC 10646[1]
Please fill all the sections A, B and C below.
Please read Principles and Procedures Document (P & P) from http://www.dkuug.dk/JTC1/SC2/WG2/docs/principles.html **for guidelines and details before filling this form.**
Please ensure you are using the latest Form from http://www.dkuug.dk/JTC1/SC2/WG2/docs/summaryform.html**.**
See also http://www.dkuug.dk/JTC1/SC2/WG2/docs/roadmaps.html **for latest *Roadmaps*.**

A. Administrative

1. **Title:** ***Proposal to Encode Two Mathematical Symbols***
2. Requester's name: *Laurențiu Iancu, Microsoft Corporation*
3. Requester type (Member body/Liaison/Individual contribution): *Individual contribution*
4. Submission date: *2009-July-28*
5. Requester's reference (if applicable):
6. Choose one of the following:
   This is a complete proposal: *Yes*
   (or) More information will be provided later:

B. Technical – General

1. Choose one of the following:
   a. This proposal is for a new script (set of characters): *No*
      Proposed name of script:
   b. The proposal is for addition of character(s) to an existing block: *Yes*
      Name of the existing block: *Miscellaneous Mathematical Symbols-A*
2. Number of characters in proposal: *2*
3. Proposed category (select one from below - see section 2.2 of P&P document):

A-Contemporary	X	B.1-Specialized (small collection)		B.2-Specialized (large collection)	
C-Major extinct		D-Attested extinct		E-Minor extinct	
F-Archaic Hieroglyphic or Ideographic				G-Obscure or questionable usage symbols	

4. Is a repertoire including character names provided?
   a. If YES, are the names in accordance with the "character naming guidelines" in Annex L of P&P document? *Yes*
   b. Are the character shapes attached in a legible form suitable for review? *Yes*
5. Who will provide the appropriate computerized font (ordered preference: True Type, or PostScript format) for publishing the standard? *The author*
   If available now, identify source(s) for the font (include address, e-mail, ftp-site, etc.) and indicate the tools used:
6. References:
   a. Are references (to other character sets, dictionaries, descriptive texts etc.) provided? *Yes*
   b. Are published examples of use (such as samples from newspapers, magazines, or other sources) of proposed characters attached? *Yes*
7. Special encoding issues:
   Does the proposal address other aspects of character data processing (if applicable) such as input, presentation, sorting, searching, indexing, transliteration etc. (if yes please enclose information)? *Yes*
   Proposed UCD character properties are included.
8. Additional Information:

Submitters are invited to provide any additional information about Properties of the proposed Character(s) or Script that will assist in correct understanding of and correct linguistic processing of the proposed character(s) or script. Examples of such properties are: Casing information, Numeric information, Currency information, Display behaviour information such as line breaks, widths etc., Combining behaviour, Spacing behaviour, Directional behaviour, Default Collation behaviour, relevance in Mark Up contexts, Compatibility equivalence and other Unicode normalization related information. See the Unicode standard at http://www.unicode.org for such information on other scripts. Also see http://www.unicode.org/Public/UNIDATA/UCD.html and associated Unicode Technical Reports for information needed for consideration by the Unicode Technical Committee for inclusion in the Unicode Standard.

[1] Form number: N3152-F (Original 1994-10-14; Revised 1995-01, 1995-04, 1996-04, 1996-08, 1999-03, 2001-05, 2001-09, 2003-11, 2005-01, 2005-09, 2005-10, 2007-03, 2008-05)

C. Technical - Justification

1. Has this proposal for addition of character(s) been submitted before? *No*
   If YES explain
2. Has contact been made to members of the user community (for example: National Body, user groups of the script or characters, other experts, etc.)? *Yes*
   If YES, with whom? *Gerhard X Ritter (University of Florida, US), Peter Sussner (Universidade Estadual de Campinas, Brazil), Gonzalo Urcid Serrano (Instituto Nacional de Astrofísica, Óptica y Electrónica, Puebla, Mexico)*
   If YES, available relevant documents: *Supporting evidence and references in proposal*
3. Information on the user community for the proposed characters (for example: size, demographics, information technology use, or publishing use) is included? *Yes*
   Reference: *Sections 1 and 5 of proposal*
4. The context of use for the proposed characters (type of use; common or rare) *Common*
   Reference: *Sections 1 and 5 of proposal*
5. Are the proposed characters in current use by the user community? *Yes*
   If YES, where? Reference: *Scientific publications in the fields of artificial intelligence, neural networks, mathematical morphology, pattern recognition, and related areas*
6. After giving due considerations to the principles in the P&P document must the proposed characters be entirely in the BMP? *Yes*
   If YES, is a rationale provided? *Yes*
   If YES, reference: *Section 2c of proposal; proposed allocation is in a block of mathematical symbols, together with related characters of analogous UCD properties*
7. Should the proposed characters be kept together in a contiguous range (rather than being scattered)? *Yes*
8. Can any of the proposed characters be considered a presentation form of an existing character or character sequence? *No*
   If YES, is a rationale for its inclusion provided?
   If YES, reference:
9. Can any of the proposed characters be encoded using a composed character sequence of either existing characters or other proposed characters? *No*
   If YES, is a rationale for its inclusion provided?
   If YES, reference:
10. Can any of the proposed character(s) be considered to be similar (in appearance or function) to an existing character? *No*
    If YES, is a rationale for its inclusion provided?
    If YES, reference:
11. Does the proposal include use of combining characters and/or use of composite sequences? *No*
    If YES, is a rationale for such use provided?
    If YES, reference:
    Is a list of composite sequences and their corresponding glyph images (graphic symbols) provided?
    If YES, reference:
12. Does the proposal contain characters with any special properties such as control function or similar semantics? *No*
    If YES, describe in detail (include attachment if necessary)
13. Does the proposal contain any Ideographic compatibility character(s)? *No*
    If YES, is the equivalent corresponding unified ideographic character(s) identified?
    If YES, reference:

1. Background

This proposal is to encode two mathematical symbols used in scientific publications in the fields of artificial intelligence, computational neuroscience, fuzzy systems, machine learning, mathematical morphology, neural networks, pattern recognition, and related disciplines. The names and glyphs of the two symbols are shown in Figure 1.

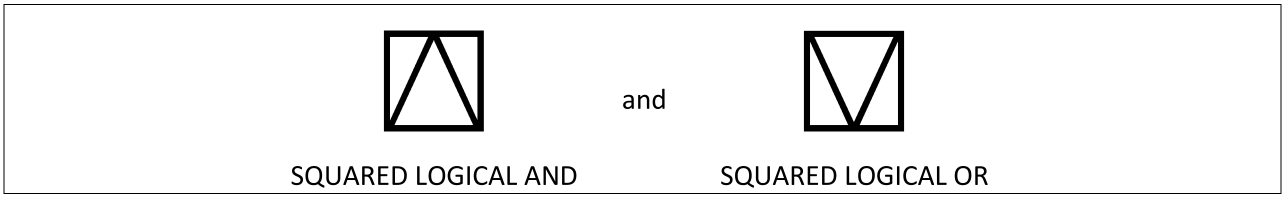

Figure 1: Enlarged glyphs of the proposed symbols.

The characters were introduced by Professor Gerhard X Ritter in the 1980s in technical reports of the USAF, DARPA, and the DoD, followed by publications of the Society of Photo-Optical Instrumentation Engineers (SPIE). The characters are first attested in [1] published in 1987. Since then, the characters have seen continuous use in specialized literature. Today they have widespread use internationally and appear in publications in Brazil, China, Greece, Japan, Mexico, Spain, the US and other countries.

The symbols denote nonlinear operations involving addition followed by minimum or maximum and are colloquially referred to as ‘box-min’ and ‘box-max.’ In mathematical morphology and morphological neural networks, they represent the erosion and dilation operators. The square enclosures were designed to indicate the distinction from convolution operators that involve multiplication followed by addition. The figures in Section 5 illustrate the symbols in a variety of contexts.

Besides standalone, the symbols also appear in combining character sequences to write other operators. For instance, with a combining tilde above, they denote fuzzy min/max product operators or XOR min/max convolution operators. These are respectively illustrated in Figures 4 and 6 and would be represented in Unicode as combining character sequences consisting of the proposed SQUARED LOGICAL AND/OR followed by U+0303 COMBINING TILDE.

2. Encoding

There are three conceivable approaches to generate the squared symbols: reuse existing characters that have similar-looking glyphs, use U+20DE COMBINING ENCLOSING SQUARE, or encode new characters. Of the three approaches, encoding new characters seems the most appropriate.

a. Reuse existing characters merely based on glyph similarity

The characters U+2353 APL FUNCTIONAL SYMBOL QUAD UP CARET and U+234C APL FUNCTIONAL SYMBOL QUAD DOWN CARET have similar glyphs with the proposed symbols. Figure 2 illustrates the APL symbols in a few fonts available in Microsoft Office and Microsoft Windows 7.

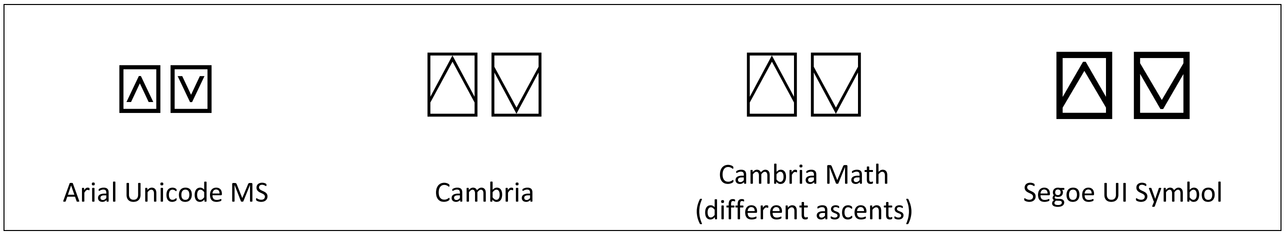

Figure 2: Glyphs of the APL functional symbols Quad Up/Down Caret (U+2353, U+234C) in a few fonts.

Although the APL symbols' glyphs are similar to those of the proposed characters, the enclosing rectangles and position of the carets are not satisfactory when compared to the morphological symbols in publications, illustrated in Section 5.

More importantly, the APL symbols are inadequate to reuse for two reasons:

- The APL characters are used for representing functional symbols of the APL programming language in Unicode;
- The UCD properties of the APL symbols are different from those of mathematical symbols, in particular the general category and bidi class.

The distinction between APL functional symbols and mathematical symbols with similar glyphs can be observed by contrasting existing Unicode characters, such as the following:

- ⍁ U+2341 APL FUNCTIONAL SYMBOL QUAD SLASH vs. ⧄ U+29C4 SQUARED RISING DIAGONAL SLASH;
- ⍂ U+2342 APL FUNCTIONAL SYMBOL QUAD BACKSLASH vs. ⧅ U+29C5 SQUARED FALLING DIAGONAL SLASH;
- ⌻ U+233B APL FUNCTIONAL SYMBOL QUAD JOT vs. ⧇ U+29C7 SQUARED SMALL CIRCLE.

The glyphs shown above are from Cambria Math. In terms of character properties, the APL symbols have General_Category = Other_Symbol and Bidi_Class = Left_To_Right, whereas the mathematical symbols have property values Math_Symbol and Other_Neutral, respectively.

b. Use U+20DE COMBINING ENCLOSING SQUARE

An alternative approach to represent the morphological symbols would be to use U+2227 LOGICAL AND and, respectively, U+2228 LOGICAL OR in combining character sequences with U+20DE COMBINING ENCLOSING SQUARE: <U+2227, U+20DE> and <U+2228, U+20DE>. However, the proposed characters are standalone entities that do not decompose into equivalent sequences. Other squared symbols in the blocks Mathematical Operators (U+229E SQUARED PLUS, …, U+22A1 SQUARED DOT OPERATOR) and Miscellaneous Mathematical Symbols-B (U+29C4 SQUARED RISING DIAGONAL SLASH, …, U+29C8 SQUARED SQUARE) do not decompose either. The proposed symbols also appear as base characters in combining sequences to form additional operators, as shown in Figures 4 and 6 in Section 5.

c. Encode new characters

Compared to the previous alternatives, encoding new characters seems the most appropriate solution for representing the proposed symbols in Unicode. The proposed allocation is at code points U+27CE and U+27CF in the space available in the Miscellaneous Mathematical Symbols-A block. In this approach, the codes U+27CE and U+27CF are used standalone to represent the plain box-min and box-max symbols. To represent the operators with tilde above, the combining character sequences are <U+27CE, U+0303> and <U+27CF, U+0303>.

The characters have UCD properties analogous to other squared mathematical symbols already encoded, as described in the following section.

3. Character Names, Properties and Glyphs

The proposed names for the two symbols are SQUARED LOGICAL AND and SQUARED LOGICAL OR, respectively. The names describe the elements that the symbols comprise to give the characters general purpose as mathematical symbols rather than tie them to specific semantics. The names were chosen by similarity with other existing characters (e.g., U+229E SQUARED PLUS, U+229F SQUARED MINUS, U+22A0 SQUARED TIMES, U+22A1 SQUARED DOT OPERATOR, U+29C6 SQUARED ASTERISK, etc.) and in accordance with Unicode naming conventions.

The UCD properties of the proposed symbols are analogous to those of related characters, as given in the following table.

UnicodeData.txt entry (gc, ccc, bc, dt, Bidi_M, etc.)	Script	Line_Break	Math
27CE;SQUARED LOGICAL AND;Sm;0;ON;;;;;N;;;;;	Common (Zyyy)	Alphabetic (AL)	Yes
27CF;SQUARED LOGICAL OR;Sm;0;ON;;;;;N;;;;;	Common (Zyyy)	Alphabetic (AL)	Yes

Table 1: UCD properties of the proposed characters (Math = Y derives implicitly from gc = Sm).

The glyphs of the two symbols consist of a logical AND and, respectively, OR enclosed within a square. The glyphs that have been used in print have the ends of the AND and OR coinciding with the vertices of the square and the tip touching the middle of the opposite edge of the square, as shown in Figure 1. These shapes may be the result of the tools that have been used to create the symbols, since encoded characters were not available. Nevertheless, these shapes have been used systematically and nowadays can be considered representative for the two symbols. As infix operators, the symbols are normally typeset with additional side spacing.

If the proposed characters are accepted for encoding, it is recommended that their glyphs be harmonized in the code charts with the other squared mathematical symbols.

4. Code Charts

UniBook code charts are appended at the end of this document. UniBook source project files and a font will be made available to the editors.

5. Supporting Evidence

Hence, W_{XY} is the least upper bound of all perfect recall memories involving the $\boxed{\vee}$ operation and M_{XY} is the greatest lower bound of all perfect memories involving the $\boxed{\wedge}$ operation. Furthermore, if there exist perfect recall memories, then the canonical memories are also perfect recall memories.	$\mathbf{z}^\gamma = M_{ZZ} \boxed{\wedge} \mathbf{z}^\gamma \leq M_{ZZ} \boxed{\wedge} \mathbf{x}^\gamma \leq \mathbf{x}^\gamma$. (40) In view of Theorem 5.1 and Eq. (40) we have $\mathbf{x}^\gamma = W_{XX} \boxed{\vee} \mathbf{z}^\gamma = W_{XX} \boxed{\vee} (M_{ZZ} \boxed{\wedge} \mathbf{z}^\gamma) \leq W_{XX} \boxed{\vee} (M_{ZZ} \boxed{\wedge} \mathbf{x}^\gamma) \leq W_{XX} \boxed{\vee} \mathbf{x}^\gamma = \mathbf{x}^\gamma$. (41)

Figure 3: Samples from [2] p. 98 and p. 106, illustrating the two symbols inline plain text and in mathematical expressions.

Let $\mathbf{y} \in \{0,1\}^n$. We define $W\tilde{\boxed{\vee}}\mathbf{y}$, the *fuzzy max product* of W and $\mathbf{y}$, and $M\tilde{\boxed{\wedge}}\mathbf{y}$, the *fuzzy min product* of M and $\mathbf{y}$, as follows: $(W\tilde{\boxed{\vee}}\mathbf{y})_i = \mathcal{SP}(\mathbf{y}, -\mathbf{w}_i^t)$, (56)	inal patterns [39]. AMMs are given in terms of the matrix-vector products "$\boxed{\vee}$" and "$\boxed{\wedge}$". We described the operations $\boxed{\vee}$ and $\boxed{\wedge}$ in terms of set operations for the binary case. We employed fuzzy set theory to define the operations "$\tilde{\boxed{\vee}}$" and "$\tilde{\boxed{\wedge}}$".

Figure 4: Samples from [3] p. 89 and p. 91, illustrating the two symbols inline plain text and as base characters supporting diacritical marks such as U+0303 COMBINING TILDE.

therefore the following bounds on the output patterns hold $\forall\xi; W_{XY} \boxed{\vee} \mathbf{x}^\xi \leq \mathbf{y}^\xi \leq M_{XY} \boxed{\wedge} \mathbf{x}^\xi$, that can be rewritten $W_{XY} \boxed{\vee} X \leq Y \leq M_{XY} \boxed{\wedge} X$. A matrix A is a $\boxed{\vee}$-perfect ($\boxed{\wedge}$-perfect) memory for (X,Y) if $A \boxed{\vee} X = Y$ ($A \boxed{\wedge} X = Y$). It can be proven that if A and B are $\boxed{\vee}$-perfect and $\boxed{\wedge}$-perfect memories, resp., for (X,Y), then W_{XY} and M_{XY} are also $\boxed{\vee}$-perfect and $\boxed{\wedge}$-perfect, resp.:

where $\times$ is any of the $\boxed{\vee}$ or $\boxed{\wedge}$ operators. Here $\boxed{\vee}$ and $\boxed{\wedge}$ denote the max and min matrix product, respectively defined as follows:

$$C = A \boxed{\vee} B = [c_{ij}] \Leftrightarrow c_{ij} = \bigvee_{k=1..n} \{a_{ik} + b_{kj}\}, \quad (4)$$

$$C = A \boxed{\wedge} B = [c_{ij}] \Leftrightarrow c_{ij} = \bigwedge_{k=1..n} \{a_{ik} + b_{kj}\}. \quad (5)$$

by $W \boxed{\vee} M_X^Z$ and we denote the memory given by Eq. (34) by $M \boxed{\vee} W_X^S$. Note, however, that in general $(W \boxed{\vee} M_X^Z) \boxed{\vee} \mathbf{x} \neq W \boxed{\vee} (M_X^Z \boxed{\wedge} \mathbf{x})$ and $(M \boxed{\wedge} W_X^S) \boxed{\wedge} \mathbf{x} \neq M \boxed{\vee} \times (W_X^S \boxed{\vee} \mathbf{x})$. Before we analyze $M \boxed{\vee} W_X^S$ for different choices of	*following statements are true. The matrix S is a dual kernel for (X,Y) and $M_{SY} \boxed{\wedge} (W_X^S \boxed{\vee} X) = Y$. For all $\mathbf{x} \in \{0,1\}^n$, the pattern $M_{SY} \boxed{\wedge} (W_X^S \boxed{\vee} \mathbf{x})$ is a $\wedge$-clause in $\mathbf{x}^1, \ldots, \mathbf{x}^k$ or equal to $M_{SY} \boxed{\wedge} \mathbf{0}$ or $M_{SY} \boxed{\wedge} \mathbf{1}$. Furthermore, the following*
products [6,7]. A vector $\boldsymbol{x} \in \mathbb{R}^n_{\pm\infty}$ is called a *max fixed point of A* if $A \boxed{\vee} \boldsymbol{x} = \boldsymbol{x}$ and a *min fixed point of A* if $A \boxed{\wedge} \boldsymbol{x} = \boldsymbol{x}$. An $n \times n$ matrix A is	P1. $W_{XX} \boxed{\vee} \boldsymbol{x}^\xi = \boldsymbol{x}^\xi = M_{XX} \boxed{\wedge} \boldsymbol{x}^\xi$, $\forall \xi \in K$. P2. $W_{XX} \boxed{\vee} \boldsymbol{x} = \boldsymbol{x}$ if and only if $M_{XX} \boxed{\wedge} \boldsymbol{x} = \boldsymbol{x}$.
D, as the expression $D \boxed{\vee} D \boxed{\vee} \ldots \boxed{\vee} D$, where the "$\boxed{\vee}$"-symbol occurs $r-1$ times. Note that the operation "$\boxed{\vee}$" is associative	$B \boxed{\vee} \mathbf{x} \leq \mathbf{c}$ and is called the *principal solution* [8]. Using the isotonicity of the $\boxed{\vee}$-product, we conclude that $B \boxed{\vee} \mathbf{x}^\sharp$ is the

Figure 5: Additional samples from various sources illustrating the two symbols.
From the top down: [4] p. 529; [5] p. 881; [6] p. 630; [7] p. 2103; [8] p. 561 and p. 562.

and

$$\mathbf{b} = \mathbf{a} ⟎ \mathbf{t}\,,$$

where

$$\mathbf{b}(\mathbf{y}) = \bigwedge_{\mathbf{x} \in \mathbf{X} \cap S_{\infty}(\mathbf{t_y})} [\mathbf{a}(\mathbf{x}) +' \mathbf{t_y}(\mathbf{x})]\,.$$

In order to distinguish between these two types of lattice transforms, we call the operator ⟏ the *morphological max convolution operator* and ⟎ the *morphological min convolution operator*. It follows from our earlier discussion that if $\mathbf{X} \cap S_{-\infty}(\mathbf{t_y}) = \varnothing$, then the value

right xor max convolution product

$$\mathbf{a} \tilde{⟏} \mathbf{t} = \left\{ (\mathbf{y}, \mathbf{b}(\mathbf{y})) \,:\, \mathbf{b}(\mathbf{y}) = \bigvee_{\mathbf{x} \in \mathbf{X}} [\mathbf{a}(\mathbf{x}) \tilde{+} \mathbf{t_y}(\mathbf{x})]\,,\ \mathbf{y} \in \mathbf{Y} \right\}$$

right xor min convolution product

$$\mathbf{a} \tilde{⟎} \mathbf{t} = \left\{ (\mathbf{y}, \mathbf{b}(\mathbf{y})) \,:\, \mathbf{b}(\mathbf{y}) = \bigwedge_{\mathbf{x} \in \mathbf{X}} [\mathbf{a}(\mathbf{x}) \tilde{+}' \mathbf{t_y}(\mathbf{x})]\,,\ \mathbf{y} \in \mathbf{Y} \right\}$$

7.2 Basic Morphological Operations: 191
Boolean Dilations and Erosions

The image algebra formulation of the dilation of the image **a** by the structuring element **B** is given by

$$\mathbf{b} := \mathbf{a} ⟏ \mathbf{t}.$$

The image algebra equivalent of the erosion of **a** by the structuring element **B** is given by

$$\mathbf{b} := \mathbf{a} ⟎ \mathbf{t}^*.$$

[29]. Apart from the previously defined matrix-products called *additive maximum* ("⟏") and *additive minimum* ("⟎"), min-

Figure 6: Samples from [9] (p. 27, p. 28, p. 191) and [10] (p. 805) illustrating various operators expressed with the proposed symbols, alone and in combining character sequences. The second sample uses U+0303 COMBINING TILDE.

6. References

[1] G. Ritter, M. Shrader-Frechette, P. Gader, "Image Algebra: Final Report, Period 1984–1985," Technical Report AD-A176 152, AFATL-TR-86-79, Defense Technical Information Center, Defense Logistics Agency, 1987.

[2] G. Ritter, G. Urcid, L. Iancu, "Reconstruction of Patterns from Noisy Inputs Using Morphological Associative Memories," Journal of Mathematical Imaging and Vision, volume 19, issue 2, pp. 95–111, Kluwer Academic Publishers, 2003, ISSN 0924-9907.

[3] P. Sussner, "Generalizing Operations of Binary Autoassociative Morphological Memories Using Fuzzy Set Theory," Journal of Mathematical Imaging and Vision, volume 19, issue 2, pp. 81–93, Kluwer Academic Publishers, 2003, ISSN 0924-9907.

[4] M. Graña, B. Raducanu, P. Sussner, G. Ritter, "On Endmember Detection in Hyperspectral Images with Morphological Associative Memories," Lecture Notes in Computer Science, volume 2527, Advances in Artificial Intelligence – IBERAMIA 2002, pp. 526–535, Springer, 2003, ISBN 978-3-540-00131-7, ISSN 0302-9743.

[5] I. Villaverde, M. Graña, A. d'Anjou, "Morphological Neural Networks and Vision Based Mobile Robot Navigation," Lecture Notes in Computer Science, volume 4131, Artificial Neural Networks – ICANN 2006, pp. 878–887, Springer, 2006, ISBN 978-3-540-38625-4, ISSN 0302-9743.

[6] P. Sussner, "Associative Morphological Memories Based on Variations of the Kernel and Dual Kernel Methods," Neural Networks, volume 16, issues 5–6, pp. 625–632, Elsevier, 2003, ISSN 0893-6080.

[7] G. Ritter, G. Urcid, M. Schmalz, "Autonomous Single-pass Endmember Approximation Using Lattice Auto-associative Memories," Neurocomputing, volume 72, issues 10–12, pp. 2101–2110, Elsevier, 2009, ISSN 0925-2312.

[8] P. Sussner, M. Valle, "Gray-Scale Morphological Associative Memories," IEEE Transactions on Neural Networks, volume 17, issue 3, pp. 559–570, IEEE Computational Intelligence Society, 2006, ISSN 1045-9227.

[9] G. Ritter, J. Wilson, "Handbook of Computer Vision Algorithms in Image Algebra," 2nd edition, CRC Press LLC, 2001, ISBN 0-8493-0075-4.

[10] P. Sussner, M. Valle, "Implicative Fuzzy Associative Memories," IEEE Transactions on Fuzzy Systems, volume 14, issue 6, pp. 793–807, IEEE Computational Intelligence Society, 2006, ISSN 1063-6706.

	27C	27D	27E
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
A			
B			
C			
D			
E	⟎ 27CE		
F	⟏ 27CF		

Operators

27CE ⟎ SQUARED LOGICAL AND
= box-min
• morphological min product operator
• morphological erosion operator
• additive minimum operator

27CF ⟏ SQUARED LOGICAL OR
= box-max
• morphological max product operator
• morphological dilation operator
• additive maximum operator