# Four-Dimensional Multi-Inkdot Finite Automata

YASUO UCHIDA, and TAKAO ITO \* Ube National College of Technology

Dept. of Business Administration 2-14-1, Tokiwadai Ube City JAPAN 755-8555 uchida@ube-k.ac.jp and ito@ube-k.ac.jp HIDENOBU OKABE Nittetsu Hitachi Systems Engineering, Inc. Financial Solutions Division 8-1, Akashi-Chou, Chuou-Ku, Tokyo JAPAN 104-6591 okabe.hidenobu@live.jp

MAKOTO SAKAMOTO, HIROSHI FURUTANI, and MICHIO KONO University of Miyazaki Faculty of Engineering 1-1, Gakuen Kibanadai Nishi, Miyazaki City JAPAN 889-2192 sakamoto@cs.miyazaki-u.ac.jp

*Abstract:* During the past about thirty-five years, many automata on a two- or three-dimensional input tape have been proposed and a lot of properties of such automata have been obtained. On the other hand, we think that recently, due to the advances in computer animation, motion image processing, and so forth, it is very useful for analyzing computational complexity of multi-dimensional information processing to explicate the properties of four-dimensional automata, i.e., three-dimensional automata with the time axis. In this paper, we propose a four-dimensional multi-inkdot finite automaton and mainly investigate its recognizability of four-dimensional connected pictures. Moreover, we briefly investigate some basic accepting powers of four-dimensional multi-inkdot finite automata.

Key-Words: Alternation, Connected Pictures, Finite Automaton, Four-Dimension, Inkdot, Recognizability

# **1** Introduction

Related to the historical open problem of whether deterministic and nondeterministic space (especially lower-level) complexity classes are separated, inkdot Turing machines were introduced in [43]. An inkdot machine is a conventional Turing machine capable of dropping an inkdot on a given input tape for a landmark, but unable to further pick it up. Against an earlier expectation, it was proved that nondeterministic inkdot Turing machines are more powerfull than nondeterministic ordinary Turing machines for sublogarithmic space bounds [5,11,20,23-26,39]. As is well-known in the theory of automata on a two-dimensional input tape [2,3,6,7,9,22,34], there is a set of square tapes accepted by a nonderministic finite automaton but not by any deterministic Turing machine with sublogarithmic space bounds[15-19]. Thus, it makes no sense to ask the question

of whether the separation exists between deterministic and nondeterministic complexity classes for the two-dimensional Turing machines with sublogarithmic space bounds [33,56-59,63]. On the other hand, there is another important aspect in the inkdot mechanism : we can consider a two-dimensional finite automaton with inkdots as a weak recognizer of the inherent properties of digital pictures[35-37,46,62]. By this motivation, in [27], two-dimensional finite automaton with inkdots was introduced as a weak recognizer of the inherent properties of digital pictures, rather than a two-dimensional Turing machine supplied with a one-dimensional working type. In [27], it is proved that two-dimensional deterministic multi-inkdot automata are eqivalent to ordinary twodimensional finite automata, i.e., there exists no hierarchy based on the number of inkdots for the twodimensional deterministic case[32]. The basic properties of two-dimensional versions of nondeterministic inkdot automata, alternating inkdot automata which have only universal states, and general alternating inkdot automata were also investigated in [27]. By

<sup>\*</sup>Takao Ito is a visiting research professor of School of Management, New Jersey Institute of Technology in USA from March 29, 2008 to March 28, 2009.

the way, the question of whether processing threedimensional digital patterns is much difficult than two-dimensional ones is of great interest from the theoretical and practical standpoints[60,64-66]. From this point of view, we introduced a three-dimensional finite automaton with inkdots as a weak recognizer of the inherent properties of three-dimensional digital pictures, rather than a three-dimensional Turing machine supplied with a one-dimensional working type [50]. We also emphasize the point that inkdot automaton is a restricted version of "marker (or pebble) automaton", which can pick up the marker put down previously on the input tape [39]. In recent years, due to the rapid development of modern technologies such as computer animation, motion image processing, virtual reality systems and so on, it has become increasingly apparent that the study of multi-dimensional pattern processing has started to play a crucial role in our society [1,4,42,54,55]. Thus, we think that it is very useful for analyzing computation of multi-dimensional pattern processing to explicate the properties of four-dimensional automata, i.e., three-dimensional automata with the time axis [28,51]. In this paper, we introduce a four-dimensional multi-inkdot finite automaton, and mainly investigate its ability to recognize the topological properties of four-dimensional binary connected pictures in Section 4 [16]. Moreover, in Section 3, we briefly investigate some basic properties of fourdimensional multi-inkdot finite automata.

# 2 Preliminaries

**Definition 2.1.** Let  $\Sigma$  be a finite set of symbols. A four-dimensional tape over  $\Sigma$  is a four-dimensional rectangular array of elements of  $\Sigma$  [44,47]. The set of all four-dimensional tapes over  $\Sigma$  is denoted by  $\Sigma^{(4)}$ . Given a tape  $x \in \Sigma^{(4)}$ , for each  $j(1 \le j \le 4)$ , we let  $l_j(x)$  be the length of x along the  $j^{th}$  axis. When  $1 \le i_j \le l_j(x)$  for each  $j(1 \le j \le 4)$ , let  $x(i_1, i_2, i_3, i_4)$  denote the symbol in x with coordinates  $(i_1, i_2, i_3, i_4)$ , as shown in Fig. 1.

Furthermore, we define

$$x[(i_1, i_2, i_3, i_4), (i'_1, i'_2, i'_3, i'_4)],$$

when  $1 \le i_j \le i'_j \le l_j(x)$  for each integer  $j(1 \le j \le 4)$ , as the four-dimensional tape y satisfying the following:

- (i) for each  $j(1 \le j \le 4)$ ,  $l_j(y) = i'_j i_j + 1$ ;
- (ii) for each  $r_1$ ,  $r_2$ ,  $r_3$ ,  $r_4$   $(1 \le r_1 \le l_1(y), 1 \le r_2 \le l_2(y), 1 \le r_3 \le l_3(y), 1 \le r_4 \le l_4(y)), y(r_1, r_2, r_3, r_4) = x(r_1 + i_1 1, r_2 + i_2 1, r_3 + i_3 1, r_4 + i_4 1).$



Fig. 1: Four-Dimensional Input Tape.

Now, we introduce four-dimensional multi-inkdot finite automaton.

**Definition 2.2.** Let k be a non-negative integer. A four-dimensional alternating k-inkdot automaton (4-AIk) is a septuple

$$M = (Q, U, q_0, F, \Sigma, \{0, 1\}, \delta),$$

where

- (1) Q is a finite set of *states*,
- (2)  $U \subseteq Q$  is a set of universal states,
- (3)  $q_0 \in Q$  is the *initial state*,
- (4)  $F \subseteq Q$  is the set of accepting states,
- (5)  $\Sigma$  is a finite *input alphabet*,
- (6) {0,1} is the presence and absence signs of inkdots,

For any  $q, q' \in Q, a, a' \in \Sigma, u = (u_1, ..., u_k), u' = (u'_1, ..., u'_k), v = (v_1, ..., v_k), v' = (v'_1, ..., v'_k) \in \{0, 1\}^k$ and  $d \in \delta$ , if  $((q', u'), (a', v'), d) \in ((q, u), (a, v))$  then (i) a = a' and (ii) for each i  $(1 \le i \le k), (u_i, v_i, u'_i, v'_i) \in \{(1, 0, 1, 0), (1, 0, 0, 1), (0, 0, 0), (0, 1, 0, 1)\}.$ 

That is, a 4-AIk M is a four-dimensional alternating finite automaton (4-AFA) with k dots of

Yasuo Uchida, Takao Ito and Hidenobu Okabe

ink, as shown in Fig. 2. Thus a 4-AIk is a fourdimensional alternating finite automaton [49,51] capable of dropping an inkdot on at most k tape-cells of the input for a landmark, once on each cell, but incapable of picking it up or erasing it. A state qin Q-U is said to be *existential*. The machine Mhas a read-only rectangular input tape with boundary symbols "#"'s. If the input head falls off the boundaries of the input tape, then the machine M can make no further move. An element of  $Q \times \{0,1\}^k$  is called an extended state. An element of  $\Sigma \times \{0,1\}^k$  is called an extended input symbol,  $(\Sigma \times \{0,1\}^k$  itself is called an extended input alphabet). An extended state (q, u) represents the situation that M is in state q and M holds or does not hold the  $i^{th}$  inkdot in the finite control, according to the value of  $u_i$  which is equal to 1 when M holds the inkdot. An extended input symbol (a, v) represents the situation that the input symbol on the current cell is a and the  $i^{th}$  inkdot exists in the same cell, according to the value  $v_i$  which is equal to 1 when the inkdot exists. A step of Mconsists of reading one symbol from the input tape, moving the input head in specified direction  $d \in \{$ east,west,south,north,up,down,future,past,no move }, and extering a new state, in accordance with the next move relation. In this paper, we assume that reader is familiar with the concept of alternation [8,29-31,40,41,48]. In necessary, see [3].



Fig. 2: Four-Dimensional Multi-Inkdot Finite Automaton.

**Definition 2.3.** Let  $M = (Q, U, q_0, F, \Sigma, \{0, 1\}, \delta)$  be a 4-Alk. An *extended input tape*  $\tilde{x}$  of M is a

four-dimensional tape obtained from the original tape x such that (1)  $\tilde{x} \in (\Sigma \times \{0,1\}^k)^{(4)}$ , (2)  $l_1(\tilde{x}) = l_1(x)$ ,  $l_2(\tilde{x}) = l_2(x)$ ,  $l_3(\tilde{x}) = l_3(x)$  and  $l_4(\tilde{x}) = l_4(x)$ , and (3) for each  $i_1$ ,  $i_2$ ,  $i_3$ ,  $i_4$   $(1 \le i_1 \le l_1(\tilde{x}),$  $1 \le i_2 \le l_2(\tilde{x})$ ,  $1 \le i_3 \le l_3(\tilde{x})$ ,  $1 \le i_4 \le l_4(\tilde{x})$ ),  $\tilde{x}(i_1,$  $i_2, i_3, i_4) = (x(i_1, i_2, i_3, i_4), v)$ , where  $v \in \{0,1\}^k$ . The *initial input tape*  $x^0$  of M is an extended input tape  $\tilde{x}$  such that for each  $i_1, i_2, i_3, i_4$   $(1 \le i_1 \le l_1(\tilde{x}),$  $1 \le i_2 \le l_2(\tilde{x}), 1 \le i_3 \le l_3(\tilde{x}), 1 \le i_4 \le l_4(\tilde{x})), \tilde{x}(i_1, i_2, i_3,$  $i_4) = (x(i_1, i_2, i_3, i_4), 0)$ , where 0 = (0, 0, ..., 0).

**Definition 2.4.** Let  $M = (Q, U, q_0, F, \Sigma, \{0, 1\}, \delta)$  be a 4-AIk. A configuration of M on x is an element of

$$(\Sigma \times \{0,1\}^k)^{(4)} \times (Q \times \{0,1\}^k) \times \boldsymbol{N}^4,$$

where N is the set of all natural numbers. The first component of configuration  $c=(\tilde{x},(q,u),(i_1, i_2, i_3, i_4))$  is an extended input tape of M. The second component of c is an extended state of M. The third component of c is the input head position of M. If q is the state associated with configuration c, then cis said to be the *universal* (existential, accepting) configuration if q is a universal (existential, accepting) state. The *initial* configuration of M on x is  $I_M(x)=(x^0,(q_0,1),(1,1,1,1))$ , where 1=(1,1,...,1).

**Definition 2.5.** Given  $M = (Q, U, q_0, F, \Sigma, \{0, 1\}, \delta)$ , we write

 $c \vdash_M c'$ 

and say c' is a *successor* of c if configuration c' follows from configuration c in one step of M, according to the transition rules  $\delta$ .  $\vdash *_M$  denotes the reflexive transitive closure of  $\vdash$ . A *computation path* of M on x is a sequence

$$c_0 \vdash_M c_1 \vdash_M \cdots \vdash_M c_n \ (n \ge 1).$$

A *computation tree* of M is a nonempty labeled tree with the properties,

- (1) each node  $\pi$  of the tree is labeled tree with a configuration  $l(\pi)$ ,
- (2) if  $\pi$  is an internal node (a nonleaf) of the tree,  $l(\pi)$  is universal and

$$\{c \mid l(\pi) \vdash_M c\} = \{c_1, ..., c_k\},\$$

then  $\pi$  has exactly k children  $\rho_1,...,\rho_k$  such that  $l(\rho_i)=c_i$ ,

(3) if  $\pi$  is an internal node of the tree and  $l(\pi)$  is existential, then  $\pi$  has exactly one child  $\rho$  such that

$$l(\pi) \vdash_M l(\rho).$$

An accepting computation tree of M on x is a finite computation tree whose root is labeled with  $I_M(x)$ and whose leaves are all labeled with accepting configurations. We say that M accepts x if there is an accepting computation tree of M on input x.

**Definition 2.6.** Let M be a 4-NIk and x be an input tape for M. Suppose that M accepts x and uses up all of the k inks. Then, there exists an accepting computation path  $P_M(x)=c_0 \vdash_M c_1 \vdash_M \cdots \vdash_M c_f$   $(f \ge 0)$ , where  $c_0$  is the initial configuration and  $c_f$  is the accepting configuration of M on x. From  $P_M(x)$ , we can extract a sequence of k different extended input tapes  $x^0, x^1, \dots, x^k$ , where  $x^0$  is the initial extended input tape of M.

By "4-DIk" ("4-NIk", "4-UIk") we denote a fourdimensional deterministic k-inkdot finite automaton (a four-dimensional nondeterministic k-inkdot finite automaton, a four-dimensional alternating k-inkdot finite automaton with only universal states). A fourdimensional alternating finite automaton is a special case of four-dimensional alternating multi-inkdot automaton, i.e. it is equivalent to 4-AI0. By "4-AF"("4-DF","4-NF", "4-UF") we denote a four-dimensional alternating finite automaton (a four-dimensional deterministic finite automaton, a four-dimensional nondeterministic finite automaton, a four-dimensional alternating finite automaton with only universal states). Let T[M] be the set of four-dimensional tapes accepted by a machine M, and the class of sets accepted by 4-AIk's is defined as follows:

 $\pounds$ [4-AIk]={ $T \mid T = T(M)$  for some 4-AIk M}.

 $\pounds$ [4-DIk],  $\pounds$ [4-NIk],  $\pounds$ [4-UIk],  $\pounds$ [4-AF],  $\pounds$ [4-DF],  $\pounds$ [4-NF],  $\pounds$ [4-UF] and etc. are defined in the same way as  $\pounds$ [4-AIk].

At the end of this section, we give set-theoritic notations involved in our discussion. For as set T of four-dimensional tapes, the complement of T is denoted by  $\overline{T}$ . Define co- $\pounds = \{\overline{T} \mid T \in \pounds\}$ .

# **3** Basic Properties

In this section, we briefly deal with some basic accepting powers of four-dimensional multi-inkdot finite automata. We first investigate the properties of deterministic version of inkdot automata, and show that deterministic inkdot automata are equivalent to ordinary deterministic finite automata. That is, no hierarchy exists based on the number of inkdots for deterministic case.

**Theorem 3.1.**  $\pounds[4\text{-}DF] = U_{k>1}\pounds[4\text{-}DIk].$ 

**Proof:** The simulation method of Ref.[43], which only treats one-ink machines, is also valid in our fourdimensional case. We recall this technique for the beginning: 4 - DF M' behaves in the same way as  $4 \text{-} DI_1$ M until it uses its own ink. When M will drop its ink, M' memorizes the input symbol on the current cell without ink drop. After that, M' continue to simulate M, except when M' encounters the same input symbol as in the finite control. When this happens, M' memorizes the current state of M and performs depth-first backward search [43] of the computation of M toward the initial configuration to test whether the encountered cell was the place of ink drop. If M' reaches the initial configuration of M, it then performs forward simulation of M until it will drop the ink. After that, M' continue to simulate M from the memorized state as if there is an ink at this position. Based on the idea above, we will prove the theorem by induction on the number of inkdots. That is, assuming that 4-DF's can simulate 4-DIk's, we will show that a 4-DIk M' can simulate the given 4-DI(k+1)M. Before M drops the k + 1st ink, when each time M' encounters any of the previously dropped inks, it memorize the pair of the ink name and the current state to use just like the initial configuration in the case of k = 1 described above. If M would drop the k + 1st ink, M' memorizes the current input symbol in the finite control. After that, M' continues to simulate M, except when M' encounters the same input symbol as the recorded symbol in the finite control. In this case, M' performs depth-first backward search and forward simulation of M between the current configuration and the most recent configuration that encountered the inkdots represented by the newest pain in the finite control.

Next, we investigate the properties of nondeterministic version of inkdot automata, and show that nondeterministic 1-inkdot automata are more powerful than ordinary nondeterministic finite automata.

**Lemma 3.1.** Let  $T_1 = \{x \in \{0,1\}^{(4)} \mid \exists m \geq 1 \ [l_1(x) = l_2(x) = l_3(x) = l_4(x) = 2m \\ \& \forall i_1, \forall i_2, \forall i_3, \forall i_4 \ (1 \leq i_1 \leq 2m, 1 \leq i_2 \leq 2m, 1 \leq i_3 \leq 2m, 1 \leq i_4 \leq m) [x(i_1, i_2, i_3, i_4) = x(i_1, i_2, i_3, i_4 + m)]]\}$ . Then (1)  $\overline{T_1} \in \mathcal{L}[4\text{-}NI1]$ , (2)  $\overline{T_1} \notin \mathcal{L}[4\text{-}AF]$ .

**Proof:** (1) a 3- $NI_1$  M accepting  $T_1$  acts as follows. Given an input tape x with  $l_1(x) = l_2(x) = l_3(x) = l_4(x) = 2m$ , M firstly scans the  $i_4$ <sup>th</sup> cubic array from the top plane to the bottom plane in a cubic array, from the northmost row to the southmost row in a plane, and from the westmost cell to the eastmost cell in a row. This sequence of moves is called a *systematic*  scan. M guesses some position  $(i_1, i_2, i_3, i_4)$  such that  $x(i_1, i_2, i_3, i_4) \neq x(i_1, i_2, i_3, i_4 + m)$  in the systematic scan. Then, it drops the ink at  $(i_1, i_2, i_3, i_4)$ and goes to the southeastmost boundary of the  $i_4$ <sup>th</sup> cubic array, i.e. the position  $(2m+1, 2m+1, 2m+1, i_4)$ in the systematic scan. Next, M moves its input head down by using information of number 'm' until M reaches the northwestmost boundary of the top plane of the (k + m)th cubic array, i.e. the position (0, 0, 0, k + m). Then, while going to the southeastmost boundary of the (k+m)th cubic array in the systematic scan, it existentially branches two machines, one of which continues to go to the southeastmost boundary, and the other of which memorizes the input symbol 'b' on the current position and goes upward to check the inequality  $b \neq (i_1, i_2, i_3, i_4)$ . If this check succeeds, then it enters an accepting state. Otherwise, it halts in a non-accepting state. It is clear that  $T(M) = T_1$ . (2) By using the same idea in the proof of part (2) of Lemma 5.1 in [43], we can easily show that  $T_1$  is not in  $\pounds[4\text{-}AF]$ .

From Lemma 3.1 and the trivial fact  $\pounds[4-NF] \subseteq \pounds[4-AF]$ , we can get

**Theorem 3.2.**  $\pounds[4-NF] \subsetneq \pounds[4-NI1]$ .

We next investigate the properties of alternating inkdot automata which have only universal states.

**Theorem 3.3.**  $\pounds[4-UF^c] \subsetneq \pounds[4-UI1].$ 

**Proof:** In the proof of Part (1) of Lemma 3.1, the constructed 4-*NI*1 *M* accepting  $\overline{T}_1$  always halts, i.e., it never enters a loop for any time for any input. Based on this fact and by exchanging the existential states of *M* to universal states, non-accepting states to accepting states, and accepting states to non-accepting halting states, we can obtain a 4-*UI*1 accepting  $T_1$ . On the other hand, from the trivial fact  $\pounds[4-UF] \subseteq \pounds[4-$ *AF*] and Part (2) of Lemma 3.1, we get the desired result  $T_1 \notin \pounds[4-UF^c]$ .

**Theorem 3.4.** co- $\pounds[4$ - $UIk] \subseteq \pounds[4$ -NI(k + 1)], for each  $k \ge 0$ .

**Proof:** Without loss of generality, we assume that, when a 4-*UIk* M wants to reject a given input x, it always enters a loop. From this assumption, we can say that M does not accept x iff there exists a computation path  $P_M(x)$  such that  $c_0 \vdash_M c_1 \vdash_M \cdots c \vdash_M \cdots \vdash_M c$ , where  $c_0$  is the initial configuration and the cycle  $c \vdash_M \cdots \vdash_M c$  represents a loop. (Given a M, we write  $c \vdash_M c'$  and say c' is a *successor* of c if con-

figuration c' follows from configuration c in one step of M, according to the *transaction rules*.) To simulate the complement of M, a 4-NI(k + 1) guesses  $P_M(x)$  by using its existential states and halts in a non-accepting state when M will enter an accepting state. M' uses k inks for the trace of the prefix subpath  $c_0 \vdash_M \cdots \vdash_M c$  in which M consumes at most kinks. The k + 1st ink of M' is used for the detection of the loop  $c \vdash_M \cdots \vdash_M c$ . Note that M' never enters an accepting state if any computation of M has no loop.

**Corollary 3.1.** co- $\pounds[4$ - $UF] \subseteq \pounds[4$ -NI1].

Finally, we investigate the properties of alternating inkdot automata which can use both universal states and existential states of finite control. We have

**Theorem 3.5.** co- $\pounds[4$ - $AIk] \subseteq \pounds[4$ -AI(k + 1)], for each  $k \ge 0$ .

**Proof:** Without loss of generality, we assume that, when a 4-AIk M rejects the given input x, it always enters a loop. From this, we can say that M does not accept x iff there exists a non-accepting computation tree which has no leaf, i.e. any path descendant from the root of which enters an infinite loop. At each step of M, a 4-AI(k + 1) M' existentially branches into two machines, one of which drops an ink for detecting a loop starting from the current configuration of M. Both machines then continue further simulation of M under the conditions such that the universal and existential states of M are exchanged with each other in M'. Of course, when M halts in an accepting state, M' halts in a non-accepting state. If there exists a non-accepting (infinite) computation tree of M, then there exists an accepting finite computation tree of M'. Conversely, if there exists an accepting computation (finite) tree of M, then there exists a nonaccepting finite computation tree of M'. Note that M'uses k inks until M enters a loop and uses one ink to verify the existence of a loop.

**Corollary 3.2.**  $\operatorname{co} \mathscr{L}[4-NF] \cup \operatorname{co} \mathscr{L}[4-UF] \subseteq \mathscr{L}[4-AI1].$ 

## 4 Recognizability

Before proceeding to the main subject, we need the following definition.

**Definition 4.1.** Let x be a four-dimensional tape over  $\{0,1\}^{(4)}$ . A 1-component of x is the maximal subset P of  $N \times N \times N \times N$  satisfying the following :

- (i) for each  $(i_1, i_2, i_3, i_4) \in P$ ,  $1 \le i_1 \le l_1(x)$ ,  $1 \le i_2 \le l_2(x)$ ,  $1 \le i_3 \le l_3(x)$ ,  $1 \le i_4 \le l_4(x)$  and  $x(i_1, i_2, i_3, i_4) = 1$ ;
- (ii) for any  $(i_1, i_2, i_3, i_4)$ ,  $(i'_1, i'_2, i'_3, i'_4) \in P$ , there exists a sequence  $(i_{1.0}, i_{2.0}, i_{3.0}, i_{4.0})$ ,  $(i_{1.1}, i_{2.1}, i_{3.1}, i_{4.1})$ ,...,  $(i_{1.n}, i_{2.n}, i_{3.n}, i_{4.n})$  of elements in P such that  $(i_{1.0}, i_{2.0}, i_{3.0}, i_{4.0}) = (i_1, i_2, i_3, i_4)$ ,  $(i_{1.n}, i_{2.n}, i_{3.n}, i_{4.n}) = (i'_1, i'_2, i'_3, i'_4)$  and  $|i_{1.j} - i_{1.j-1}| + |i_{2.j} - i_{2.j-1}| + |i_{3.j} - i_{3.j-1}| + |i_{4.j} - i_{4.j-1}| \leq 1$   $(1 \leq j \leq n)$ .

A tape  $x \in \{0, 1\}^{(4)}$  is called *connected* if there exists exactly one 1-component of x. We denote the set of all the four-dimensional connected tapes by  $T_c$  [49].

**Lemma 4.1.** Let M be a 4-NIk and Q be the set of states of M. Let x be an input tape for M. Suppose that M accepts x and uses up all of the k inks. For any accepting computation path  $P_M(x)$  of M on x, the number of visits to the position  $(i_1, i_2, i_3, i_4)$  on  $x^l$  of  $P_M(x)$  is at most |Q| for any  $l(0 \le l \le k)$  and for any  $(i_1, i_2, i_3, i_4)$   $[0 \le i_1 \le l_1(x) + 1, 0 \le i_2 \le l_2(x) + 1, 0 \le i_3 \le l_3(x) + 1, 0 \le i_4 \le l_4(x) + 1]$ .

**Proof :** Suppose that, on some  $x^l$  of  $P_M(x) = c_0, c_1$ , ..., $c_f$ , M visits some position  $(i_1, i_2, i_3, i_4)$  more than |Q| times. Then, there exists a configuration cin  $P_M(x)$  such that  $c_0 \vdash_M \cdots c \vdash_M \cdots c \vdash_M \cdots \vdash_M$  $c_f$ . By removing the subpath (a cycle) from c, up to c, we get another valid accepting computation path  $c_0$  $\vdash_M \cdots c \vdash_M \cdots \vdash_M c_f$  which has no repetition of c.

In the Lemma below, we will show that the fourdimensional inkdot finite automata can be simulated by "four-dimensional nondeterministic on-line tessellation acceptors (4-NOTA)". The 4-NOTA is a restricted type of cellular automaton in which cells do not make transitions at every time step : rather, a transition "wave" passes once diagonally across the array. We say that a 4-NOTA M accepts an input x if the resulting state of the southmost and eastmost bottom automaton (accepting cell) of the cellular space is a predetermined accepting state [12]. To simplify the proof, we adopt "four-dimensional nondeterministic multipass on-line tessellation acceptor (4-NMPOTA)" instead of 4-NOTA itself [13]. A 4-NMPOTA is a multi-wave version of 4-NOTA. In each pass, the 4-NMPOTA outputs a fourdimensional tape consisting of the states of the finite state machines resulting from the current transition wave on the cellular space and regards it as the input in the next pass. The 4-NMPOTA repeats such passes until its accepting cell enters an accepting state. In this paper, we will use cellular automata such as 4-NOTA and 4-NMPOTA, but omit their rigorous definitions. If necessary, see [12,13].

It is known that two-dimensional nondeterministic multipass on-line tessellation acceptors whose number of passes are restricted to some constant k have the same power as ordinary twodimensional nondeterministic on-line tessellation acceptors [10,13,14,21,61]. This result can be extended to the four-dimensional case as following lemma, where let 4-NMPOTA(k) be a four-dimensional nondeterministic multipass on-line tessellation acceptors whose number of passes are restricted to some constant k.

Lemma 4.2.  $\pounds$ [4-NOTA]= $\cup_{k\geq 1}\pounds$ [4-NMPOTA(k)].

**Lemma 4.3.**  $\cup_{k>0} \pounds$ [4-NIk]  $\subseteq \pounds$ [4-NOTA].

**Proof** : Let M be a 4-NIk and x be an input for M. From Lemma 3.1, we can assume that for any  $l(1 \le l \le k)$ , the number of visits of M to any position  $(i_1, i_2, i_3, i_4)$  of x' is bounded by a constant. We construct a 4-NMPOTA M' with k+1 passes accepting T(M) as follows : At the first pass M' simulates M from the initial configuration to the configuration in which M drops the first ink (of course, the position of the drop must be guessed simultaneously). In general, at the  $l^{th}$  pass, M' simulates M from the configuration in which M drops the  $(l-1)^{st}$  ink to the cinfiguration in which M drops the  $l^{th}$  ink. At the last  $(k+1)^{st}$  pass, M' simulates M from the configuration in which M drops the  $k^{th}$  ink to an accepting configuration (if any). In each pass, the simulation is done by the same method as in the proof of the fact that two-dimensional nondeterministic finite automata can be simulated by two-dimensional nondeterministic on-line tesellation acceptors described in [12]. It is clear that T(M')=T(M). From this and Lemma 4.2, the lemma follows. 

**Theorem 4.1.**  $T_c \notin \bigcup_{k>1} \pounds$  [4-NIk].

**Proof :** It is shown in [38] that  $T_c$  is not recognizable by three-dimensional nondeterministic on-line tessellation acceptors. By using the same technique, we can get the fact that  $T_c \notin \pounds$  [4-NOTA]. It is obvious from the fact and Lemma 4.3 that the theorem holds.  $\Box$ 

**Note 4.1.** From Theorem 4.1 and the fact that  $T_c$  is recognizable by a three-dimensional alternating finite automaton [49], it follows that  $\pounds$ [4-NIk]  $\subsetneq \pounds$ [4-AIk] for any integer k.

We then show that the set  $T_c$  is recognizable by a

4-UI1.

### **Theorem 4.2.** $T_c \in \pounds$ [4-UI1].

**Proof**: We can get the desired result by parallelizing the raster-scan method of deterministic one marker automaton described in [43]. Roughly speaking, A 4-UI1 M checks whether the symbols "1"'s in each row on each plane of a given input  $x \in \{0,1\}^{(4)}$  are all connected : To this end, it must be checked whether any two consecutive runs of "1"'s in each row on each plane are connected to each other. Let P and Q be the last (eastmost) point of a run and the first (westmost) point of the next run, respectively. When Mreaches P, it universally branches into two machines, one of which goes on to check the connectedness of the next two consecutive runs, and the other of which acts as follows. It first drops the ink there (point P) and then follows the border  $\beta$  defined by P and the symbol "0" immediately succeeding it. At each move around  $\beta$ , if M is at a point Q' that has a "0" on its west, it scans to the west until it hits a symbol "1" (or reaches a boundary symbol "#"). If this position has the ink on it, then M knows that Q' is Q, so that Q is connected to P, and thus it enters an accepting state. Otherwise, it returns to Q' and resumes the border following. If the border following is completed and it reaches the ink at P again, it knows that Q is not connected to P, so that the "1"'s in x is not connected, then it halts in a non-accepting state. The remaining task that "1"'s in two consecutive rows on each plane and two censecutive planes in a given input are connected each other is done in the same way as the verification described above of the connectedness of pairs of consecutive runs of each row.  $\square$ 

Finally, we show that the set  $\overline{T}_c$  is recognizable by a 4-NI1.

**Theorem 4.3.**  $\overline{T}_c \in \pounds$ [4-NI1].

**Proof :** In the previous theorem, the constructed 4-UI1 M accepting  $T_c$  always halts, i.e., it never enters a loop for any time for any input. Based on this and by exchanging the universal states of M for existential states, non-accepting states for accepting states and accepting states for non-accepting halting states, we can obtain a 4-NI1 accepting  $\overline{T_c}$ .

Corollary 4.1.  $\bar{T}_c \in \pounds$  [4-NOTA].

# **5** Conclusions

In this paper, we mainly investigated the recognizability of four-dimensional connected pictures by alternating multi-inkdot finite automata, and showed some properties of them. In Section 3, we briefly showed some basic accepting powrers of four-dimensional multi-inkdot finite automata. In Section 4, we show some abilities to recognize the topological properties of four-dimensional connected pictures. It will be interesting to investigate how much space is necessary and sufficient for four-dimensional alternating (nondeterministic, deterministic) Turing machines to simulate four-dimensional alternating (nondeterministic, deterministic) multi-inkdot finite automata, and the recognizability about the topological properties of digital pictures such as the interlocking component which is a chainlike connectivity. We would like to hope that some unsolved open problems concering this paper will be explicated in the near future.

#### References:

- [1] Beauchemin S. and Barron J., The computation of optical flow, *ACM Computing Surveys*, Vol27, No.3, 1996, pp.433-467.
- [2] Blum M. and Hewiit C., Automata on a two-dimensional type, *IEEE Symposium on Switching and Automata Theory*, 1967, pp.155-160.
- [3] Chandra A.K., Kozen D. C. and Stockmeyer L. J., Alternation, *Journal of the Association* for Computing Machinery, Vol.28, No.1, 1981, pp.114-133.
- [4] Foley J. D., van Dam A., Feiner S. K. and Hughes J. F., *Computer Graphics : Principles* and Practice 2nd Edition in C, Addison-Wesley Publishing Company, Inc., 1996.
- [5] Geffert V., Nondeterministic computations in sublogarithmic space and space constructibility, *SIAM Journal on Computing*, Vol.20, No.3, 1991, pp.484-498.
- [6] Hopcroft J. E. and Ullman J. D., Some results on tape-bounded Turing machines, *Journal of the ACM*, Vol.19, No.2, 1972, pp.283-295.
- [7] Hopcroft J. E. and Ullman J. D., Introduction to Automata Theory, Languages, and Computation, *Addison-Wesley, Reading, Mass.*, 1979.
- [8] Hromkovič J., On the power of alternation in automata theory, *Journal of Computer and System Sciences*. Vol.31, No.1, 1985, pp.28-39.
- [9] Ibarra O. H. and Melson R. T., Some results concerning automata on two-dimensional tapes, *International Journal of Computer Mathematics*, Vol.4, No.A, 1974, pp.269-279.

- [10] Inoue K. and Nakamura A., An n-dimensional on-line tessellation acceptors(in Japanese), *The Transaction s of the IECE*, Vol.J59-D, No.4, 1976, pp.229-236.
- [11] Inoue K. and Nakamura A., Some properties of two-dimensional nondeterministic finite automata and parallel sequential array acceptors(in Japanese), *The Transaction s of the IECE*, Vol.J60-D, No.11, 1977, pp.990-997.
- [12] Inoue K. and Nakamura A., Some properties of two-dimensional on-line tessellation acceptors, *Information Sciences*, Vol.13, 1977, pp.95-121.
- [13] Inoue K. and Nakamura A., Two-dimensional multipass on-line tessellation acceptors, *Information and Control*, Vol.41, 1979, pp.305-323.
- [14] Inoue K. and Takanami T., A note on bottom-up pyramid acceptors, *Information Processing Letters*, Vol.8, No.1, 1979, pp.34-37.
- [15] Inoue K. and Takanami T., Some properties of three-way two-dimensional multicounter automata over square tapes(in Japanese), *The Transaction s of the IECE*, Vol.J62-D, No.10, 1979, pp.673-680.
- [16] Inoue K., Takanami I. and Taniguchi H., Threeway two-dimensional simple multihead finite automata – hierachical properties(in Japanese), *The Transaction s of the IECE*, Vol.J62-D, No.2, 1979, pp.65-72.
- [17] Inoue K., Takanami I. and Taniguchi H., Threeway two-dimensional simple multihead finite automata – closure properties(in Japanese), *The Transaction s of the IECE*, Vol.J62-D, No.4, 1979, pp.273-280.
- [18] Inoue K. and Takanami I., Three-way tapebouded two-dimensional Turing machines, *Information Sciences*, Vol.17, 1979, pp.195-220.
- [19] Inoue K. and Takanami I., Three-way twodimensional multicounter automata, *Information Sciences*, Vol.19, No.1, 1979, pp.1-20.
- [20] Inoue K., Takanami I. and Taniguchi H., Twodimensional alternating Turing machines, *Theoretical Computer Sciences*, Vol.27, 1983, pp.61-83.
- [21] Inoue K., Takanami I. and Nakamura A., Connected pictures are not recognizable by deterministic two-dimensional on-line tessellation acceptors, *Computer Vision, Graphics, Image Processing*, Vol.26, 1984, pp.126-129.
- [22] Inoue K. and Takanami I., A survey of twodimensional automata theory, *Information Sciences*, Vol.55, 1991, pp.99-121.

- [23] Inoue K., Ito A. and Takanami I., A relationship between nodeterministic Turing machines and 1inkdot Turing machines with small space, *Information Processing letters*, Vol.43, 1992, pp.225-227.
- [24] Ito A., Inoue K., Takanami I. and Taniguchi H., Two-dimensional alternating Turing machines with only universal states, *Information and Control*, Vol.55, No.1-3, 1982, pp.193-221.
- [25] Ito A., Inoue K. and Takanami I., A note on three-way two-dimensional alternating Turing machines, *Information Sciences*, Vol.45, 1988, pp.1-22.
- [26] Ito A., Inoue K. and Takanami I., Complexity of acceptance problems for two-dimensional automata, a perspective in Theoretical Computer Science, Series in Computer Science, Vol.16, 1989, pp.70-94.
- [27] Ito A., Inoue K., Takanami I. and Wang Y., The effect of inkdots for two-dimensional automata, *International Journal of Pattern Recognition and Artificial Intelligence*, Vol.9, No.5, 1995, pp.777-796.
- [28] Ito T., Sakamoto M., Tomozoe N., Iihoshi K., Furutani H., Kono M., Tamaki T. and Inoue K., Two topics concerning three-dimensional synchronized alternating Turing machines, WSEAS Transactions on Computers, Issue 10, Vol.5, 2006, pp.2149-2155.
- [29] King K. N., Alternating multihead finite automata, *Theoretical Computer Science*, Vol.61, 1985, pp.149-174.
- [30] King K. N., Measures of parallelism in alternating computation trees, *Proceedings of the 13th ACM Symposium on Theory of Computing*, 1981, pp.189-201.
- [31] Ladner R. E., Lipton R. J. and Stockmeyer L. J., Alternating pushdown automata, *Proceedings* of the 19th IEEE Symposium on Foundations of Computer Science, 1978, pp.92-106.
- [32] Makino T., Okabe H., Taniguchi S., Sakamoto M. and Inoue K., Three-dimensional multiinkdot automata, *International Journal of Artificial Life and Robotics*, Vol.9, No.2, 2005, pp.99-101.
- [33] Michel P., A survey of space complexity, *Theoretical Computer Science*, Vol.101, 1992, pp.99-132.
- [34] Morita K., Computational complexity in oneand two-dimensional tape automata, *Ph.D Thesis Osaka University*, 1978.

- [35] Nagamizu Y., Sakamoto M., Ito T., Iihoshi K., Tomozoe N., Furutani H., Kono M., Ikeda S., Tamaki T. and Inoue K., Remarks on recognizability of topological components by threedimensional automata, *Proceedings of the 12th International Symposium on Artificial Life and Robotics*, Oita, Japan, 2007, pp.135-139.
- [36] Nakamura A. and Aizawa K., On the recognition of properties of three-dimensional pictures, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.PAMI-7, 1985, pp.708-713.
- [37] Nakamura A. and Aizawa K., Detection of interlocking components in three-dimensional digital pictures, *Information Sciences*, Vol.40, 1986, pp.143-153.
- [38] Nakamura A. and Rosenfeld A., Threedimensional connected pictures are not recognizable by finite-state acceptors, *Technical Report TR*-566, *Computer Science Center*, *University of Maryland*, 1991.
- [39] Okabe H., Sakamoto M. and Inoue K., Computational Complexity for the Simulation of Four-Dimensional Marker Automata by Four-Dimensional Turing Machines, *Proceedings* of the 8<sup>th</sup> International Symposium on Artificial Life and Robotics, , 2003, pp.114-117.
- [40] Paul W. J., Prauss E. J. and Reischuk R., On alternation; *Acta Informatica*, Vol.14, 1980, pp.243-255.
- [41] Paul W. J., Prauss E. J. and Reischuk R., On alternation; *Acta Informatica*, Vol.14, 1980, pp.391-403.
- [42] Peng E. and Li L., Modeling human skeleton proportions from monocular data, *WSEAS Transactions on Computers*, Issue 3, Vol.5, 2006, pp.681-687.
- [43] Ranjan D., Chang R. and Hartmanis J., Space bounded computations : Review and New Separation Results, *Theoretical Computer Science*, Vol.80, 1991, pp.289-302.
- [44] Rosenfeld A. and Milgram D. L., Parallel / sequential array automata, *Information Processing Letters*, Vol.2, 1973, pp.43-46.
- [45] Rosenfeld A. and Kak A. C., *Digital Picture Processing*, Academic Press, New York, 1976.
- [46] Rosenfeld A., *Picture Languages–Formal Models for Picture Recognition*, Academic Press, Inc., 1979.
- [47] Rosenfeld A. and Milgram D. L., Parallel / sequential array automata, *Information Processing Letters*, Vol.2, 1973, pp.43-46.

- [48] Ruzzo W. L., Tree-size bounded alternation, Journal of Computer and System Sciences, Vol.21, 1980, pp.218-235.
- [49] Sakamoto M., Inoue K. and Takanami I., A Note on Three-Dimensional Alternating Turing Machines with Space Smaller than log m, *Information Sciences*, Vol.72, 1993, pp.225-249.
- [50] Sakamoto M., Inoue K. and Ito A., Basic Properties of Three-Dimensional Multi-Inkdot Automata, Proceedings of the 1996 IEICE General Comference - Information and Systems 1, 1997, p.5.
- [51] Sakamoto M., Okabe H., Nogami S., Taniguchi S., Makino T., Nakama Y., Saito M., Kono M. and Inoue K., A note on four-dimensional finite automata, *WSEAS Transactions on Computers*, Issue 5, Vol.3, 2004, pp.1651-1656.
- [52] Shah A. N., Pebble automata on arrays, *Computer Graphics Image Processing*, Vol.3, 1974, pp.236-246.
- [53] Shah A. N., Pushdown automataon arrays, *Information Sciences*, Vol.25, 1981, pp.175-193.
- [54] Sherman W. R. and Craig A. B., Undestanding Virtual Reality – Interface, Application, and Design, Morgan Kaufmann Publishers, 2003.
- [55] Smith A. T. and Snowden R. J., eds. Visual Detection of Motion, Academic Press, Inc., 1994.
- [56] Sudborough I. H., Efficient algorithms for paths system programs and applications to alternating and time-space complexity classes, *Proceedings* of the 21st IEEE Symposium on Foundations of Computer Science, 1980, pp.62-73.
- [57] Szepietowski A., There are no fully space constructible functions between log log *n*, *Information Processing Letters*, Vol.24, 1987, pp.361-362.
- [58] Szepietowski A., On three-way two-dimensional multicounter automata, *Information Sciences*, Vol.55, 1991, pp.35-47.
- [59] Szepietowski A., On space functions constructed by two-dimensional Turing machines, *Information Sciences*, Vol.60, 1992, pp.177-183.
- [60] Taniguchi H., Inoue K. and Takanami I., A note on three-dimensional finite automata, *Information Sciences*, Vol.26, 1982, pp.65-85.
- [61] Taniguchi H., Inoue K. and Takanami I., *k*-neigh-borhood template *A*-type 2-dimensional bounded cellular acceptor (in Japanese), *The Transactions of the IEICE*, Vol.J69-D, No.3, 1986, pp.291-301.
- [62] Tsao Y. F. and Fu K. S., A parallel thinning algorithm for 3-D pictures, *Computer Graphics Image Processing*, Vol.17, 1981, pp.315-331.

- [63] Turing A. M., On computable number, with an application to the Entscheidungsproblem, *Proceedings of the London Math. Soc.*, Vol.2, No.42, 1936, pp.230-265.
- [64] Umeo H., Morita K. and Sugata K., Pattern recognition by automata on a two-dimensional tapes(in Japanese), *The Transactions of the IE-ICE*, Vol.J59-D, No.11, 1976, pp.817-824.
- [65] Yamamoto H. and Noguchi S., Time- and -leaf bounded l-tape alternating Turing machines(in Japanese), *The Transactions of the IE-ICE*, Vol.J68-D, No.10, 1985, pp.1719-1726.
- [66] Yamamoto Y., Morita K. and Sugata K., Space complexity for recognizing connected patterns in a three-dimensional tape (in Japanese), *Technical Report of the IEICE*, No.AL79-104, 1979, pp.91-96.