F. Acar Savacı (Ed.)

LNAI 3949

# Artificial Intelligence and Neural Networks

14th Turkish Symposium, TAINN 2005
Izmir, Turkey, June 2005
Revised Selected Papers



Springer

F. Acar Savacı (Ed.)

# Artificial Intelligence and Neural Networks

14th Turkish Symposium, TAINN 2005
Izmir, Turkey, June 16-17, 2005
Revised Selected Papers





Springer

Series Editors

Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Volume Editor

F. Acar Savacı
Izmir Institute of Technology
Electrical-Electronics Engineering
Gülbahçe, Urla, Izmir, Turkey
E-mail: acarsavaci@iyte.edu.tr

Library of Congress Control Number: 2006929223

CR Subject Classification (1998): I.2, I.5, F.4.1, H.3, F.1

LNCS Sublibrary: SL 7 – Artificial Intelligence

ISSN 0302-9743
ISBN-10 3-540-36713-6 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-36713-0 Springer Berlin Heidelberg New York



Springer is a part of Springer Science+Business Media

springer.com


Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11803089 06/3142 5 4 3 2 1 0

Lecture Notes in Artificial Intelligence 3949
Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

# Preface

The Turkish Artificial Intelligence and Neural Network Symposium (TAINN) is an annual meeting where scientists present their new ideas and algorithms on artificial intelligence and neural networks with either oral or poster presentation. The TAINN-Turkish Conference on AI and NN Series started in 1992 at Bilkent University in Ankara, envisioned by various researchers in AI and NN then at the Bilkent, Middle East Technical, Boğaziçi and Ege universities as a forum for local researchers to get together and communicate. Since then, TAINN has been held annually around early summer. This year the 14th TAINN conference was organized by the EE and CE departments of the İzmir Institute of Technology with an emphasis on international contributions.

Among the 75 papers, 41 were accepted for oral presentation and 12 for poster presen-tation. In addition to the presentations, invited lectures were given by Burhan Türkşen (Knowledge/Intelligence Systems Laboratory, University of Toronto) and Joerg Siekmann (German Research Center for Artificial Intelligence DFKI, Saarland University), providing us with new developments in the field. We are very grateful to the contributions of such pioneer scientists.

The careful reviewing process by the well-recognized reviewers contributed to the quality of TAINN 2005, and therefore we thank them very much. We thank each member of the Organizing Committee and advisory board. We also thank Bora Mocan and Emre Çek, who are research assistants in the Electrical-Electronics Engineering Department of IYTE, for their efforts in the organization and in the preparation of the proceedings. We are grateful to the Turkish Scientific and Research Council of Turkey (TUBITAK) and the İzmir Branch of the Chamber of Electrical and Electronics Engineers (EMO) for their financial support.

Ferit Acar Savacı
Conference Chair

# Organizing Committee

## Chair

| | |
|---|---|
| Ferit Acar Savacı | Izmir Institute of Technology |

## Co-chair

| | |
|---|---|
| Kayhan Erciyes | Izmir Institute of Technology |

## Technical Program Chairs

| | |
|---|---|
| Bora Kumova | Izmir Institute of Technology |
| Bora Mocan | Izmir Institute of Technology |
| Deniz Çokuslu | Izmir Institute of Technology |
| Mehmet Emre Çek | Izmir Institute of Technology |

## Advisory Board

| | |
|---|---|
| Cem Say | Bogazici University |
| Cüneyt Güzeliş | Dokuz Eylul University |
| Ethem Alpaydın | Bogazici |
| H. Altay Güvenir | Bilkent |
| Kemal Oflazer | Sabancı |
| Uğur Halıcı | Middle East Technical University |

# Lecture Notes in Artificial Intelligence (LNAI)

Vol. 4068: H. Schärfe, P. Hitzler, P. Øhrstrøm (Eds.), Conceptual Structures: Inspiration and Application. XI, 455 pages. 2006.

Vol. 4065: P. Perner (Ed.), Advances in Data Mining. XI, 592 pages. 2006.

Vol. 4062: G. Wang, J.F. Peters, A. Skowron, Y. Yao (Eds.), Rough Sets and Knowledge Technology. XX, 810 pages. 2006.

Vol. 4049: S. Parsons, N. Maudet, P. Moraitis, I. Rahwan (Eds.), Argumentation in Multi-Agent Systems. XIV, 313 pages. 2006.

Vol. 4048: L. Goble, J.-J.C.. Meyer (Eds.), Deontic Logic and Artificial Normative Systems. X, 273 pages. 2006.

Vol. 4045: D. Barker-Plummer, R. Cox, N. Swoboda (Eds.), Diagrammatic Representation and Inference. XII, 301 pages. 2006.

Vol. 4031: M. Ali, R. Dapoigny (Eds.), Innovations in Applied Artificial Intelligence. XXIII, 1353 pages. 2006.

Vol. 4029: L. Rutkowski, R. Tadeusiewicz, L.A. Zadeh, J. Zurada (Eds.), Artificial Intelligence and Soft Computing – ICAISC 2006. XXI, 1235 pages. 2006.

Vol. 4027: H.L. Larsen, G. Pasi, D. Ortiz-Arroyo, T. Andreasen, H. Christiansen (Eds.), Flexible Query Answering Systems. XVIII, 714 pages. 2006.

Vol. 4021: E. André, L. Dybkjær, W. Minker, H. Neumann, M. Weber (Eds.), Perception and Interactive Technologies. XI, 217 pages. 2006.

Vol. 4020: A. Bredenfeld, A. Jacoff, I. Noda, Y. Takahashi (Eds.), RoboCup 2005: Robot Soccer World Cup IX. XVII, 727 pages. 2006.

Vol. 4013: L. Lamontagne, M. Marchand (Eds.), Advances in Artificial Intelligence. XIII, 564 pages. 2006.

Vol. 4012: T. Washio, A. Sakurai, K. Nakajima, H. Takeda, S. Tojo, M. Yokoo (Eds.), New Frontiers in Artificial Intelligence. XIII, 484 pages. 2006.

Vol. 4008: J.C. Augusto, C.D. Nugent (Eds.), Designing Smart Homes. XI, 183 pages. 2006.

Vol. 4005: G. Lugosi, H.U. Simon (Eds.), Learning Theory. XI, 656 pages. 2006.

Vol. 3978: B. Hnich, M. Carlsson, F. Fages, F. Rossi (Eds.), Recent Advances in Constraints. VIII, 179 pages. 2006.

Vol. 3963: O. Dikenelli, M.-P. Gleizes, A. Ricci (Eds.), Engineering Societies in the Agents World VI. XII, 303 pages. 2006.

Vol. 3960: R. Vieira, P. Quaresma, M.d.G.V. Nunes, N.J. Mamede, C. Oliveira, M.C. Dias (Eds.), Computational Processing of the Portuguese Language. XII, 274 pages. 2006.

Vol. 3955: G. Antoniou, G. Potamias, C. Spyropoulos, D. Plexousakis (Eds.), Advances in Artificial Intelligence. XVII, 611 pages. 2006.

Vol. 3949: F. A. Savacı (Ed.), Artificial Intelligence and Neural Networks. IX, 227 pages. 2006.

Vol. 3946: T.R. Roth-Berghofer, S. Schulz, D.B. Leake (Eds.), Modeling and Retrieval of Context. XI, 149 pages. 2006.

Vol. 3944: J. Quiñonero-Candela, I. Dagan, B. Magnini, F. d'Alché-Buc (Eds.), Machine Learning Challenges. XIII, 462 pages. 2006.

Vol. 3930: D.S. Yeung, Z.-Q. Liu, X.-Z. Wang, H. Yan (Eds.), Advances in Machine Learning and Cybernetics. XXI, 1110 pages. 2006.

Vol. 3918: W.K. Ng, M. Kitsuregawa, J. Li, K. Chang (Eds.), Advances in Knowledge Discovery and Data Mining. XXIV, 879 pages. 2006.

Vol. 3913: O. Boissier, J. Padget, V. Dignum, G. Lindemann, E. Matson, S. Ossowski, J.S. Sichman, J. Vázquez-Salceda (Eds.), Coordination, Organizations, Institutions, and Norms in Multi-Agent Systems. XII, 259 pages. 2006.

Vol. 3910: S.A. Brueckner, G.D.M. Serugendo, D. Hales, F. Zambonelli (Eds.), Engineering Self-Organising Systems. XII, 245 pages. 2006.

Vol. 3904: M. Baldoni, U. Endriss, A. Omicini, P. Torroni (Eds.), Declarative Agent Languages and Technologies III. XII, 245 pages. 2006.

Vol. 3900: F. Toni, P. Torroni (Eds.), Computational Logic in Multi-Agent Systems. XVII, 427 pages. 2006.

Vol. 3899: S. Frintrop, VOCUS: A Visual Attention System for Object Detection and Goal-Directed Search. XIV, 216 pages. 2006.

Vol. 3898: K. Tuyls, P.J. 't Hoen, K. Verbeeck, S. Sen (Eds.), Learning and Adaption in Multi-Agent Systems. X, 217 pages. 2006.

Vol. 3891: J.S. Sichman, L. Antunes (Eds.), Multi-Agent-Based Simulation VI. X, 191 pages. 2006.

Vol. 3890: S.G. Thompson, R. Ghanea-Hercock (Eds.), Defence Applications of Multi-Agent Systems. XII, 141 pages. 2006.

Vol. 3885: V. Torra, Y. Narukawa, A. Valls, J. Domingo-Ferrer (Eds.), Modeling Decisions for Artificial Intelligence. XII, 374 pages. 2006.

Vol. 3881: S. Gibet, N. Courty, J.-F. Kamp (Eds.), Gesture in Human-Computer Interaction and Simulation. XIII, 344 pages. 2006.

Vol. 3874: R. Missaoui, J. Schmidt (Eds.), Formal Concept Analysis. X, 309 pages. 2006.

Vol. 3873: L. Maicher, J. Park (Eds.), Charting the Topic Maps Research and Applications Landscape. VIII, 281 pages. 2006.

Vol. 3863: M. Kohlhase (Ed.), Mathematical Knowledge Management. XI, 405 pages. 2006.

Vol. 3862: R.H. Bordini, M. Dastani, J. Dix, A.E.F. Seghrouchni (Eds.), Programming Multi-Agent Systems. XIV, 267 pages. 2006.

Vol. 3849: I. Bloch, A. Petrosino, A.G.B. Tettamanzi (Eds.), Fuzzy Logic and Applications. XIV, 438 pages. 2006.

Vol. 3848: J.-F. Boulicaut, L. De Raedt, H. Mannila (Eds.), Constraint-Based Mining and Inductive Databases. X, 401 pages. 2006.

Vol. 3847: K.P. Jantke, A. Lunzer, N. Spyratos, Y. Tanaka (Eds.), Federation over the Web. X, 215 pages. 2006.

Vol. 3835: G. Sutcliffe, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning. XIV, 744 pages. 2005.

Vol. 3830: D. Weyns, H. V.D. Parunak, F. Michel (Eds.), Environments for Multi-Agent Systems II. VIII, 291 pages. 2006.

Vol. 3817: M. Faundez-Zanuy, L. Janer, A. Esposito, A. Satue-Villar, J. Roure, V. Espinosa-Duro (Eds.), Nonlinear Analyses and Algorithms for Speech Processing. XII, 380 pages. 2006.

Vol. 3814: M. Maybury, O. Stock, W. Wahlster (Eds.), Intelligent Technologies for Interactive Entertainment. XV, 342 pages. 2005.

Vol. 3809: S. Zhang, R. Jarvis (Eds.), AI 2005: Advances in Artificial Intelligence. XXVII, 1344 pages. 2005.

Vol. 3808: C. Bento, A. Cardoso, G. Dias (Eds.), Progress in Artificial Intelligence. XVIII, 704 pages. 2005.

Vol. 3802: Y. Hao, J. Liu, Y.-P. Wang, Y.-m. Cheung, H. Yin, L. Jiao, J. Ma, Y.-C. Jiao (Eds.), Computational Intelligence and Security, Part II. XLII, 1166 pages. 2005.

Vol. 3801: Y. Hao, J. Liu, Y.-P. Wang, Y.-m. Cheung, H. Yin, L. Jiao, J. Ma, Y.-C. Jiao (Eds.), Computational Intelligence and Security, Part I. XLI, 1122 pages. 2005.

Vol. 3789: A. Gelbukh, Á. de Albornoz, H. Terashima-Marín (Eds.), MICAI 2005: Advances in Artificial Intelligence. XXVI, 1198 pages. 2005.

Vol. 3782: K.-D. Althoff, A. Dengel, R. Bergmann, M. Nick, T.R. Roth-Berghofer (Eds.), Professional Knowledge Management. XXIII, 739 pages. 2005.

Vol. 3763: H. Hong, D. Wang (Eds.), Automated Deduction in Geometry. X, 213 pages. 2006.

Vol. 3755: G.J. Williams, S.J. Simoff (Eds.), Data Mining. XI, 331 pages. 2006.

Vol. 3735: A. Hoffmann, H. Motoda, T. Scheffer (Eds.), Discovery Science. XVI, 400 pages. 2005.

Vol. 3734: S. Jain, H.U. Simon, E. Tomita (Eds.), Algorithmic Learning Theory. XII, 490 pages. 2005.

Vol. 3721: A.M. Jorge, L. Torgo, P.B. Brazdil, R. Camacho, J. Gama (Eds.), Knowledge Discovery in Databases: PKDD 2005. XXIII, 719 pages. 2005.

Vol. 3720: J. Gama, R. Camacho, P.B. Brazdil, A.M. Jorge, L. Torgo (Eds.), Machine Learning: ECML 2005. XXIII, 769 pages. 2005.

Vol. 3717: B. Gramlich (Ed.), Frontiers of Combining Systems. X, 321 pages. 2005.

Vol. 3702: B. Beckert (Ed.), Automated Reasoning with Analytic Tableaux and Related Methods. XIII, 343 pages. 2005.

Vol. 3698: U. Furbach (Ed.), KI 2005: Advances in Artificial Intelligence. XIII, 409 pages. 2005.

Vol. 3690: M. Pěchouček, P. Petta, L.Z. Varga (Eds.), Multi-Agent Systems and Applications IV. XVII, 667 pages. 2005.

Vol. 3684: R. Khosla, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part IV. LXXIX, 933 pages. 2005.

Vol. 3683: R. Khosla, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part III. LXXX, 1397 pages. 2005.

Vol. 3682: R. Khosla, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part II. LXXIX, 1371 pages. 2005.

Vol. 3681: R. Khosla, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part I. LXXX, 1319 pages. 2005.

Vol. 3673: S. Bandini, S. Manzoni (Eds.), AI*IA 2005: Advances in Artificial Intelligence. XIV, 614 pages. 2005.

Vol. 3662: C. Baral, G. Greco, N. Leone, G. Terracina (Eds.), Logic Programming and Nonmonotonic Reasoning. XIII, 454 pages. 2005.

Vol. 3661: T. Panayiotopoulos, J. Gratch, R. Aylett, D. Ballin, P. Olivier, T. Rist (Eds.), Intelligent Virtual Agents. XIII, 506 pages. 2005.

Vol. 3658: V. Matoušek, P. Mautner, T. Pavelka (Eds.), Text, Speech and Dialogue. XV, 460 pages. 2005.

Vol. 3651: R. Dale, K.-F. Wong, J. Su, O.Y. Kwong (Eds.), Natural Language Processing – IJCNLP 2005. XXI, 1031 pages. 2005.

Vol. 3642: D. Ślęzak, J. Yao, J.F. Peters, W. Ziarko, X. Hu (Eds.), Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, Part II. XXIII, 738 pages. 2005.

Vol. 3641: D. Ślęzak, G. Wang, M. Szczuka, I. Düntsch, Y. Yao (Eds.), Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, Part I. XXIV, 742 pages. 2005.

Vol. 3635: J.R. Winkler, M. Niranjan, N.D. Lawrence (Eds.), Deterministic and Statistical Methods in Machine Learning. VIII, 341 pages. 2005.

Vol. 3632: R. Nieuwenhuis (Ed.), Automated Deduction – CADE-20. XIII, 459 pages. 2005.

Vol. 3630: M.S. Capcarrère, A.A. Freitas, P.J. Bentley, C.G. Johnson, J. Timmis (Eds.), Advances in Artificial Life. XIX, 949 pages. 2005.

Vol. 3626: B. Ganter, G. Stumme, R. Wille (Eds.), Formal Concept Analysis. X, 349 pages. 2005.

Vol. 3625: S. Kramer, B. Pfahringer (Eds.), Inductive Logic Programming. XIII, 427 pages. 2005.

¥496.00元

# Table of Contents

# A Case Study on Logging Visual Activities: Chess Game

Şükrü Ozan and Şevket Gümüştekin

Izmir Institute of Technology, Urla Izmir 35430 Turkey
sukruozan@iyte.edu.tr, sevketgumustekin@iyte.edu.tr
www.iyte.edu.tr

**Abstract.** Automatically recognizing and analyzing visual activities in complex environments is a challenging and open-ended problem. In this study this task is performed in a chess game scenario where the rules, actions and the environment are well defined. The purpose here is to detect and observe a FIDE (Fédération International des Èchecs) compatible chess board, generating a log file of the moves made by human players. A series of basic image processing operations have been applied to perform the desired task. The first step of automatically detecting a chess board is followed by locating the positions of the pieces. After the initial setup is established every move made by a player is automatically detected and verified. Intel® Open Source Computer Vision Library (OpenCV) is used in the current software implementation.

## 1 Introduction

Interpreting visual activities is an open-ended research problem in Computer Vision. The recent availability of necessary computational power to process vast amount of video data motivated research on several aspects of video processing. The temporal component of video introduces an additional dimension compared to static images enabling us extraction of meaningful events. In [1] temporal segmentation techniques are reviewed in a general terms. Several interrelated applications such as content based image retrieval [2], video annotation [3], video indexing [4] has been a major focus of research. Most of these applications are based on low level processing of pixel data. Since there is an unlimited number of visual events that need to be recognized it is impossible to expect a general purpose algorithm to extract the meaning of all the actions in a video stream. The high level information in video data can be better interpreted if the application domain is limited. Several successful studies has been carried out in sports video applications (e.g. [5][6]).

In this paper, a board game: chess has been considered as the domain of study. Compared to the other board games chess is more interesting. Besides its popularity it introduces some challenge as a vision problem with its three dimensional pieces and relatively complex rules. Most of the work on chess has been in the area of Artificial Intelligence aiming to create computer chess players. The progress of these studies ended up as a computer [7] defeating a human world champion. Other related work involves reading printed chess moves [8]. The studies similar to the one in this paper also attack the problem of interpreting a chess game in action but they are focused on

F.A. Savacı (Ed.): TAINN 2005, LNAI 3949, pp. 1 – 10, 2006.

designing and implementing robots [9][10] that can move pieces in the chess board. Here, our main concern is understanding the actions on the chess board.

By observing a standard chess board, the time instants where moves are performed by players have been detected, the moves are recognized using the visual data as well as the rules of the game and a log file of the moves is generated as a result. The camera is expected to be located with a clear view of the complete board and all pieces, but there is no further constraint on the perspective of the scene.

## 2 Chessboard Detection

A chessboard has a unique and uniform structure which makes it easily identifiable even in complex scenes. The first step of our scheme involves detection of the chessboard and extracting information about its position and orientation. A built in tool in Intel® Open Source Computer Vision Library (OpenCV) [11] is used to detect the chessboard and identify 49 inner corners.

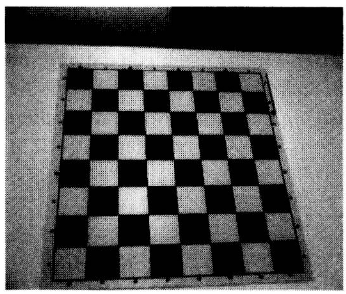


**Fig. 1.** Chess Board view example 1

The chessboard detection algorithm accepts grayscale images like the one in Figure 1. Initially, the algorithm applies a gray scale dilation operation [12]. By this operation touching corners of the squares are separated from each other. After the separation thresholding is applied to the image in order to convert the grayscale image to binary format. Generated binary image is used for a contour retrieving routine. While getting all the contour information, algorithm rejects contours with perimeters that are too small. So, contours with too small perimeters are regarded as noise in the image.

Contour analysis continues with searching for quadrangles. All the contours are checked and non-quadrangle contours are rejected. After these operations all the quadrangles and their corresponding contours in the source image are used for the next step which involves corner finding.

By using the retrieved contour plot as input, a corner finding algorithm is applied looking for sudden turns. The midpoint between the detected corners that are very close to each other are interpreted as the inner corners in the chessboard.

The 49 inner corners are verified and other corners are rejected by checking with the expected 7x7 uniform structure of the corner set.

# 3 Projection of the View

The only constraint on the position of the chessboard is that the angle between the surface normal of the board and the inverted viewing direction is sufficiently small (i.e. less than 30°). Each piece is expected to be visible. But, some undesired perspective effects are allowed. Among such effects are: slight occlusion and coverage of more than one squares by a single piece. In this section we describe how the view of the chess board is rectified.

After the inner corners of the chessboard are detected and ordered as in Figure 2, the far corner points labeled as 1,7,43 and 49 are used for the perspective projection.

**Fig. 2.** Detected and labeled corners after chessboard detection algorithm applied

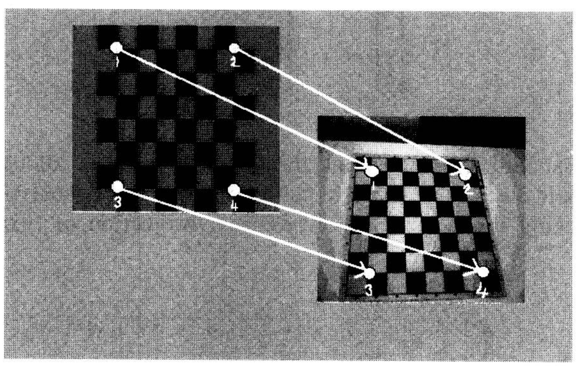


**Fig. 3.** 4 point correspondences for remapping

## 3.1 Finding the Parameters of the Perspective Transformation

A perspective projection can be formulized by the homographies [13]:

$$x_{im} = \frac{a * X + b * Y + c}{g * X + h * Y + 1}$$

$$y_{im} = \frac{d*X+e*Y+f}{g*X+h*Y+1} \tag{1}$$

Where in a backwards projection scenario (X,Y) are the coordinates on the desired projected view image and ($x_{im}$,$y_{im}$) is the coordinates on the source image. Matching 4 points (8 coordinate parameters) in the target and source images allow us to find the unknown parameters of the perspective transformation.

8 linearly independent equations can be written in matrix representation as:

$$\begin{bmatrix} X_1 & Y_1 & 1 & 0 & 0 & 0 & -X_1*x_{im1} & -Y_1*x_{im1} \\ X_2 & Y_2 & 1 & 0 & 0 & 0 & -X_2*x_{im2} & -Y_1*x_{im2} \\ X_3 & Y_3 & 1 & 0 & 0 & 0 & -X_3*x_{im3} & -Y_1*x_{im3} \\ X_4 & Y_4 & 1 & 0 & 0 & 0 & -X_4*x_{im4} & -Y_1*x_{im4} \\ 0 & 0 & 0 & X_1 & Y_1 & 1 & -X_1*y_{im1} & -Y_1*y_{im1} \\ 0 & 0 & 0 & X_2 & Y_2 & 1 & -X_1*y_{im2} & -Y_1*y_{im2} \\ 0 & 0 & 0 & X_3 & Y_3 & 1 & -X_1*y_{im3} & -Y_1*y_{im3} \\ 0 & 0 & 0 & X_4 & Y_4 & 1 & -X_1*y_{im4} & -Y_1*y_{im4} \end{bmatrix} \cdot \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} = \begin{bmatrix} x_{im1} \\ x_{im2} \\ x_{im3} \\ x_{im4} \\ y_{im1} \\ y_{im2} \\ y_{im3} \\ y_{im4} \end{bmatrix} \tag{2}$$

The 8x8 matrix can be represented as P for convenience, hence the unknown parameters can be computed as:

$$\begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} = P^{-1} \cdot \begin{bmatrix} x_{im1} \\ x_{im2} \\ x_{im3} \\ x_{im4} \\ y_{im1} \\ y_{im2} \\ y_{im3} \\ y_{im4} \end{bmatrix} \tag{3}$$

After the chessboard is detected and the projected view is calculated, players put their pieces in their places, and the orientation of the board is analyzed. This analysis is done by checking the horizontal and vertical projection of pixels to see if the

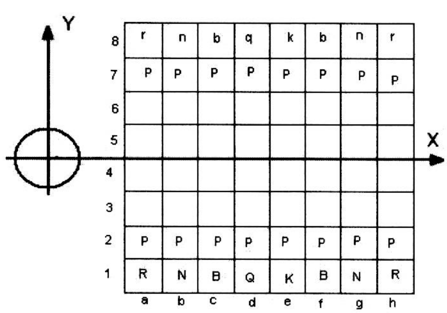



**Fig. 4.** Assumed chessboard standard, small letters are for black pieces, capital letters are for white pieces

horizontal projection shows an uneven distribution with a high number of black pixels towards the top of the board. If its orientation is not compatible with the standard given in Figure 4, then the projection is repeated as to show black pieces at the top and white pieces at the bottom in the projected view.

This is done by changing the orders of $(x_{im1},y_{im1})$, $(x_{im2},y_{im2})$, $(x_{im3},y_{im3})$ and $(x_{im4},y_{im4})$ in circular order, i.e. Instead of giving the reference points in the order:

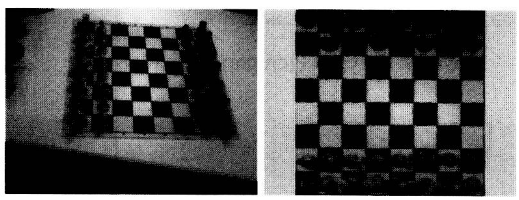
$\{ (x_{im1},y_{im1}), (x_{im2},y_{im2}), (x_{im3},y_{im3}),(x_{im4},y_{im4}) \}$

One of these sets is used:

$\{ (x_{im2},y_{im2}), (x_{im3},y_{im3}), (x_{im4},y_{im4}),(x_{im1},y_{im1}) \}$
$\{ (x_{im3},y_{im3}), (x_{im4},y_{im4}), (x_{im1},y_{im1}),(x_{im2},y_{im2}) \}$
$\{ (x_{im4},y_{im4}), (x_{im1},y_{im1}), (x_{im2},y_{im2}),(x_{im3},y_{im3}) \}$

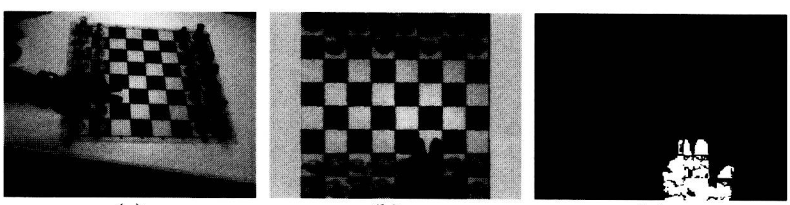
## 4 Move Detection

After the pieces are placed in order a typical scene looks like Figure 5.a. Figure 5.b is the projected view of the scene which is obtained by perspective transformation described earlier.



(a) (b)

**Fig. 5.** (a) A starting view (b) Projected view of the chess board



(a) (b) (c)

**Fig. 6.** (a)The hand in the scene (b)Projected view (c)Difference image

When the initial setup of pieces is complete the system enters in a "wait" mode for a play action. The next mode is the "alert" mode which is the state entered when a human activity is observed. The expected activity is the movement of a human hand on the chessboard. In Figure 6.a, player's hand is in the view of camera. Figure 6.b is the projected view. The difference between Figure 5.b and 6.b is shown in Figure 6.c.

A typical activity is detected from the video stream by observing the difference from the reference frame. The difference images are thresholded and number of foreground pixels are recorded. This operation forms a plot shown in Figure 7. The horizontal axis here is the time axis or frame number and the vertical axis is the number of the white pixels in difference images. This plot can easily be interpreted within the context of the chess move. An increase at the beginning of the graph indicates the beginning of action. A peak value suddenly drops to low levels when human hand stops to pick up a piece. Another peak is observed when it stops to put the piece on the board. When the initial level is reached we can assume that the move has ended.

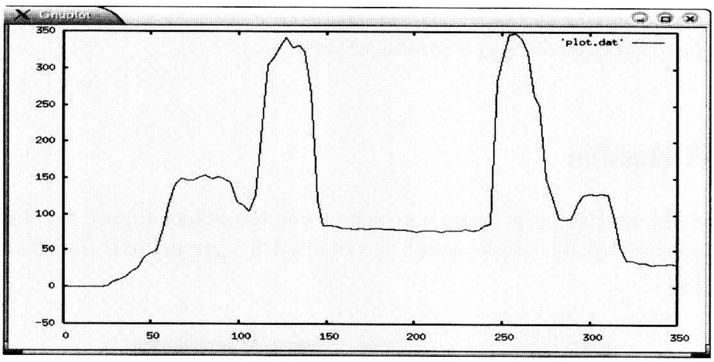



**Fig. 7.** A typical temporal plot of a move

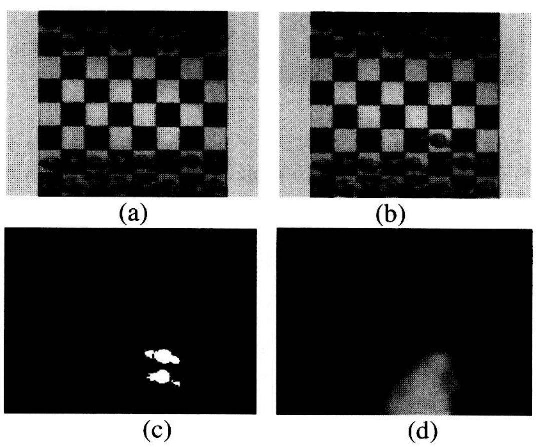



**Fig. 8.** (a) Before the move (b) After the move (c) Difference image (d) Cumulative difference

When the move is detected the system enters into the state of "analyze move". This step verifies whether or not a move is made and it determines the new placement of pieces on the board. This is done by using a cumulative difference image (Figure 8.d)