# The Focus of Attention Problem Revisited

Manjish Pal*


## Abstract

The Focus of Attention (FOA) Problem is, given a set of targets and a set of sensors in the plane, to track the targets with 'maximum possible accuracy'. The accuracy is measured in terms of the angle subtended by the sensor pairs at the assigned targets. In this paper, we consider a scenario in which we are given $n$ targets on a line $l$ and $2n$ sensors on a line $m$ such that $l \,||\, m$ and the objective is to assign sensor-pairs to the targets such that the minimum angle subtended at the targets is maximized. We give a polynomial time algorithm for a restricted version of this problem and also study some properties of the optimal solution. To our knowledge, no deterministic and exact polynomial time algorithm is known for any non-trivial version of the FOA problem when the accuracy is measured in terms of the angles subtended by the sensor pairs.


Keywords: Target tracking, Focus of Attention problem

## 1 Introduction

The Focus of Attention problem is motivated from the problem of tracking targets using sensor networks which is mainly used for the purpose of surveillance and monitoring tasks [3]. A limitation these sensors have is that one sensor is not capable of tracking a target. The sensors for example can be cameras and they can be used to estimate the position of a target. In practice, at least two such sensors are required to estimate the position of a target and for many cases like pan-tilt-zoom cameras one sensor cannot be used to track more than one target. For range sensors, three are required to localize a target [1]. In this paper, we will only consider the case in which each target can be tracked exactly by 2 cameras. To estimate the quality of tracking a good metric is required. The angle subtended by a camera pair at a target plays a crucial role in tracking the targets [4]. To have minimum uncertainty in the position of the targets the best measure is to assign the cameras to the targets such that the deviation of the subtented angle at each of the targets from 90 degrees is somewhat low. In other words if all the subtended angles are 90 degrees then the position of the targets is estimated very accurately. It follows from this that for a good tracking of a target, the angle subtended at the target should not be very small. A natural way to make sure that the no angle is very small will be to maximize the minimum angle subtended at the targets. In [1] the cameras are assumed to be on a line and the error associated with an assignment of cameras $c_i$ and $c_j$ with target $k$ is $Z_k/l_{ij}$ where $l_{ij}$ is the distance between the cameras and $Z_k$ is the distance of the cameras from the line containing the cameras. The objective is to find an assignment that minimizes the total error. Intuitively, if we fix $Z_k$ and the angle is small then the value of $l_{ij}$ will be small and error will be large. Hence this metric tries to capture the angles via an approximation. So, the natural question that arises is, can we work directly with angles and design efficient algorithms that can output an assignment that minimizes the maximum deviation from 90 degrees? Unfortunately, this problem has been shown to be intractable in [2]. Gfeller et al [2] have recently shown that given a set of $2n$ cameras and $n$ targets in the plane, it is NP-complete to decide whether there exists an assignment of cameras to track targets such that each subtended angle is 90 degrees. They also give approximation algorithms for maximizing the minimum angle and maximizing the sum of angles when cameras are placed on a line. An extensive survey and motivation on this problem is provided in [2].

To our knowledge, no deterministic and exact polynomial time algorithm is known for any non-trivial version of the FOA problem when the accuracy is measured in terms of the angles subtended by the camera pairs. In this paper we achieve such a result for a restricted version of the problem.

## 2 Problem Definition

Consider $n$ targets present on a line $l_1$ and $2n$ cameras on a line $l_2$, where $l_1$ is parallel to $l_2$. Two cameras $c_i$ and $c_j$ are said to subtend an angle $\theta$ at a target $t_k$, if $\angle c_i t_k c_j$ is $\theta$. In an **assignment**, a camera $c$ can focus on exactly one target and a target is focussed by exactly two cameras. Moreover, each camera has to be used in an assignment. Consider the following version of the

*Indian Institute of Technology Kanpur, Kanpur-208016, India, manjish@iitk.ac.in. This work was done during the CADMO internship programme-2007, at ETH-Zürich, Switzerland.

focus of attention problem:
We have to find an assignment of cameras to targets such that

- The minimum angle of all possible assignments is maximized (Parallel-MAX-MIN Problem).

Let $LHS$ denote the set of first $n$ cameras and $RHS$ denote the set of last $n$ cameras (ordered from left to right).

## 3 Parallel-MAX-MIN Problem

We start off by proving a fundamental fact regarding this optimization problem.

**Lemma 1** *There always exists an optimal solution $\mathcal{O}$ of the Parallel-MAX-MIN problem such that in $\mathcal{O}$ there are no two camera pairings $c_i, c_j$ and $c_k, c_l$ such that $i < j < k < l$ where $i, j, k, l$ denote the index of the cameras in the sorted order from left.*

**Proof** Let camera pairs $(c_i, c_j)$ and $(k, l)$ be assigned to $t_p$ and $t_q$ repectively in an optimal solution where $i < j < k < l$. If $t_p$ is to the left of $t_q$ then we can swap the assignment to get a new assignment with pairings $(c_i, c_k, t_p)$ and $(c_j, c_l, t_q)$ and in this new assignment the subtended angles for both the assignments increase, hence the minimum angle in the solution does not decrease. Similarly, if $t_p$ is to the right of $t_q$ then the pairings $(c_i, c_k, t_q)$ and $(c_j, c_l, t_p)$ will give us a solution which is optimal as well.
The following can be easily derived from the previous lemma.

**Corollary 1** *There exists an optimal solution in which each camera from the set of first $n$ cameras from left is paired with a camera in the set of last $n$ cameras.*

Consider the following decision version of the problem:
*Given an angle $\theta$ does there exist an assignment of cameras and targets such that all the subtended angles in the assignment are at least $\theta$*
It is clear that if we can solve this problem then we can solve the MAX-MIN problem just by doing a binary search on the $n^3$ possible values of $\theta$. If we choose a camera pair, $c_i$ from $LHS$ and $c_j$ from $RHS$ then there exists a **validity interval** $I$ such that for every point $p$ in the interval the angle subtented by $c_i$ and $c_j$ at $p$ is at least $\theta$. This interval is defined by the intersection of a ball $B$ with the line containing the cameras such that the boundary of this ball is that circle passing through $i$ and $j$, for which the angle subtented by the chord $ij$ at its center is $2\theta$, if $\theta$ is acute and $2\pi - 2\theta$, if $\theta$ is obtuse. So, the decision problem is equivalent to decide whether there exists an assignment of cameras and targets such that if target $t_k$ is assigned to camera pair $(i, j)$ then it lies inside the validity interval of $(i, j)$.

### 3.1 Some Properties of the Intervals

We first introduce some notations on which the rest of the paper is based. We label the first $n$ cameras from left to right as $a_1, a_2, \ldots a_n$, the last $n$ cameras from left to right as $b_n, b_{n-1}, \ldots b_1$ and the $n$ targets from left to right as $t_1, t_2, \ldots t_n$. We denote the validity interval corresponding to camera pairs $(a_i, b_j)$ for angle $\theta$ by $I_\theta(a_i, b_j)$. The words 'before' and 'after' will corresponding to ordering from left to right. We can now state the following lemmas which are easy to prove:

**Lemma 2** *If $I_\theta(a_i, b_j)$ and $I_\theta(a_k, b_l)$ are two intervals with $i \leq k$ and $j \leq l$, then $I_\theta(a_i, b_j)$ is nested within $I_\theta(a_k, b_l)$.*

**Corollary 2** *If $I_\theta(a_i, b_j)$ covers a target $t$ then it is also covered by $I_\theta(a_k, b_l)$ for all $k \leq i$ and $l \leq j$.*

### 3.2 Polynomial time Algorithm for a Restricted Case

In this section we describe the conditions that if imposed on the problem instance can bring some nice structure to the intervals defined by the cameras which can be exploited to get a greedy strategy work for the problem. We call this constraint **Interval Property** which can be stated in the following way:

*For every camera $a_i \in LHS$, $I_\theta(a_i, b_n)$ should start before the interval $I_\theta(a_{i+1}, b_1)$ and for every camera $b_j \in RHS$, the interval $I_\theta(a_n, b_j)$ should end after the interval $I_\theta(a_1, b_{j+1})$.*

Given an instance of the Parallel-MAX-MIN Problem, let $\Theta$ be the set of all angles $\theta$ such that there exist $a_i \in LHS, b_j \in RHS$ and target $t_p$ with $\angle a_i t_p b_j = \theta$. We impose the following constraint $C$ in order to make our algorithm work:
*C: For each $\theta \in \Theta$ the following should hold*
*1. All the targets are to the left of the right end-point of the interval $I_\theta(a_n, b_n)$.*
*2. Interval Property is satisfied for each angle $\theta \in \Theta$.*
Next, we derive a geometric constraint under which Interval Property is satisfied for a given value of $\theta$.

#### 3.2.1 The Height Condition

Consider the defining circles $C_i$ and $C_i{}'$ corresponding to the intervals $I_\theta(a_i, b_n)$ and $I_\theta(a_{i+1}, b_1)$. Let these circles intersect at 2 points, the one which is above the line of the cameras let it be called $X$. Let the distance of this point from the line containing the cameras be denoted by $H_{a_i}$. In order to impose the above restriction on the intervals we would like to have the distance between the parallel lines to be less than $H_{a_i}$. So, we can calculate the critical height for each $a_i$ and $b_j$ and the height $h_c$ we choose should satisfy the condition $h_c < \min\{\min_{a_i \in LHS} H_{a_i}, \min_{b_j \in RHS} H_{b_j}\}$. The following calculations are done assuming $\theta < 90°$. These can be analogously done for obtuse and right angle as well.

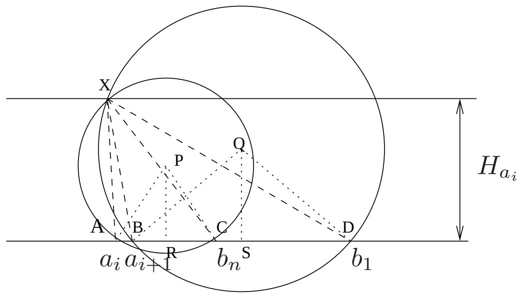


Figure 1: The Height Condition

In figure 1, $\angle AXC = \angle BXD = \theta$. Let the distance between $a_i$ and $b_n$ is $AC = d_1$, the distance between $a_{i+1}$ and $b_1$ is $BD = d_3$ and $BC = d_2$. We use coordinate geometry to find the value of $H_{a_i}$. Let the center of $C_i$ be $P$ and that of $C_i^{'}$ be $Q$. It is clear that the radius of $C_i$ is $r_1 = (d_1 + d_2)\text{cosec}\theta/2$ and that of $C_i^{'}$ is $r_2 = (d_2 + d_3)\text{cosec}\theta/2$. If the relative positions of the points $A, B, C, R, S$ is as shown in the Figure 1 then the distance between the centres of the two circles is $RS = BC - SC - BR$. Now $SC = BC - BS = d_2 - (d_2 + d_3)/2 = (d_2 - d_3)/2$ and $BR = d_2 - (d_1 + d_2)/2 = (d_2 - d_1)/2$. Therefore $RS = (d_1 + d_3)/2$. The equation of $C_i$ is $x^2 + y^2 = r_1^2$ (taking origin of coordinates at $P$) and that of $C_i^{'}$ is $(x - (d_1 + d_3)/2)^2 + (y - (d_3 - d_1)\cot\theta/2)^2 = r_2^2$. If we solve for the intersection of these circles we get

$$x^2 + y^2 - r_1^2 = (x - \frac{(d_1 + d_3)}{2})^2 + (y - \frac{(d_3 - d_1)\cot\theta}{2})^2 - r_2^2$$

$$r_1^2 - r_2^2 = \frac{(d_1 + d_2)^2}{4} + x(d_1 + d_3) + \frac{(d_3 - d_1)^2\cot^2\theta}{4} + y(d_3 - d_1)\cot\theta$$

Let,

$$A_i(\theta) = \frac{(r_1^2 - r_2^2 - (d_1 + d_3)^2/4 - (d_3 - d_1)^2\cot^2\theta/4)}{(d_1 + d_3)}$$

$$B_i(\theta) = \frac{(d_3 - d_1)\cot\theta}{(d_3 + d_1)}$$

Then we have $x = A_i(\theta) + B_i(\theta)y$. Putting this value in the equation $x^2 + y^2 = r_1^2$ we get a quadratic equation in $y$ whose positive root we denote by $y_{pos}$. Therefore, $H_{a_i} = y_{pos} + (d_1 + d_2)\cot\theta/2$. So, in order to force the aforementioned structure on the intervals we need to select $h \leq h_c$.
Before going for the algorithm we state the following lemmas which are quite easy to see and hence we omit their proofs.

**Lemma 3** *Given that interval property is satisfied, if interval $I_\theta(a_i, b_j)$ is nested by $I_\theta(a_k, b_l)$ then $k \leq i$ and $l \leq j$.*

**Lemma 4** *Given that interval property is satified, if the left end-point of $I_\theta(a_i, b_j)$ is to the left of left end-point of $I_\theta(a_k, b_l)$ and right end-point of $I_\theta(a_i, b_j)$ is to the left of right end-point of $I_\theta(a_k, b_l)$ then $i < k$ and $l < j$.*

The arrangment of the intervals satisfying interval property ($n = 3$) for a particular instance is shown in Figure 2

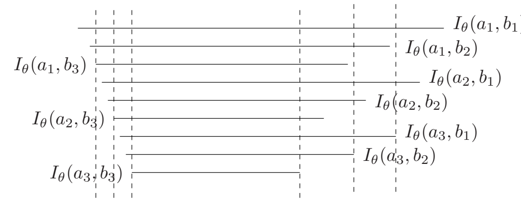


Figure 2: Arrangement of Intervals satisfying Interval Property

### 3.3 Algorithm

Now the question arises, how can this structure help us in getting a greedy strategy work. We assume a restriction on the where the cameras are placed which is as follows. We suggest a greedy strategy that works in this case. The strategy is as follows:

*Consider the targets from left to right. For the $i^{th}$ target from the left, let $I_1, I_2, \ldots I_m$ be the intervals in which it lies, assign it to the interval among these which is ending earliest. If the interval is $I_\theta(a_i, b_j)$ then remove all the intervals with left camera as $a_i$ and right camera as $b_j$.*

It can easily be seen that this algorithm can be implemented in $O(n^3 \log n)$ time.

#### Proof of Correctness of Algorithm

**Proof.** Let the given settings of cameras and targets have a solution as an assignment and let a valid assignment be called a **real assignment**. There can be many real assignments possible for an instance of the decesion problem. In this proof we will fix a real assignment (if there is any) and refer to that allthrough. A triple $(a_i, b_j, t_p)$ is said to be a valid **pairing**, if the target $t_p$ lies inside the interval defined by camera pair $a_i, b_j$. Consider the leftmost target $t_1$, and let $I_1, I_2, \ldots I_m$ be the intervals in which it lies. Also assume, $I_1 = I_\theta(a_i, b_j)$ be the interval that is finishing earliest among these. Suppose, in the real assignment, $t_1$ be assigned to the interval $I_2 = (a_k, b_l)$.

By case analysis we show that we can get a solution from the real solution which has the triple $(a_i, b_j, t_1)$ in the assignment. So, we can now remove the triple $(a_i, b_j, t_1)$ from the real assignment and the rest of the $n-1$ targets and $2(n-1)$ cameras will have a solution. The base case ($n = 2$) can be verified easily. The following cases need to be handled:

**Case 1.** $j \neq n$. If $I_1$ is one of $I_\theta(a_1, b_1), I_\theta(a_1, b_2), \ldots, I_\theta(a_1, b_{n-1})$, say $I_\theta(a_1, b_j)$, $j \leq n-1$ then only intervals that can cover it are $I_\theta(a_1, b_1), I_\theta(a_1, b_2), \ldots, I_\theta(a_1, b_j)$. So, in the real assignment it must be covered by $I_\theta(a_1, b_1)$ where $k \leq j$. If $k = j$ we are done, else if $I_\theta(a_{i_1}, b_j)$ covers some target $t_p$ then we can swap these to get new pairings as $(a_1, b_j, t_1), (a_{i_1}, b_k, t_p)$ and this will be a valid assignment because of above conditions.

**Case 2.** $j = n$. The following subcases arise:
**(a)** $k \leq i$. Let $k < i$, then because of the imposed constraint on the intervals, $I_\theta(a_k, b_l)$ will start before $I_\theta(a_i, b_j)$ and since it is ending after $I_\theta(a_i, b_j)$ it will be nesting the interval $I_\theta(a_i, b_j)$. Hence, from lemma 3 we have $j \leq l$. Now let in the real assignment we have the following 3 assignments $(a_k, b_l, t_1), (a_i, b_{j'}, t_p), (a_{i'}, b_j, t_q)$. We interchange the pairings to get 3 new pairings $(a_i, b_j, t_1), (a_k, b_{j'}, t_p), (a_{i'}, b_l, t_q)$ which are valid as $I_\theta(a_i, b_{j'})$ is nested in $I_\theta(a_k, b_{j'})$ and $I_\theta(a_{i'}, b_j)$ is nested in $I_\theta(a_{i'}, 1)$.

**(b)** $k > i$. If $k > i$, then starting point of $I_\theta(a_k, b_l)$ will be to the right of that of $I_\theta(a_i, b_j)$ and hence the intervals $I_\theta(a_i, b_j)$ and $I_\theta(a_k, b_l)$ satisfy the conditions of lemma 4, which implies $l < j = n$. Now assume that the real assignment has the triples $(a_k, b_l, t_1), (a_i, b_{j'}, t_p), (a_{i'}, b_j, t_q)$. Now, the new pairings we would like to propose are $(a_i, b_j, t_1), (a_k, b_{j'}, t_p), (a_{i'}, b_l, t_q)$. In this case the only problematic pair is $(a_i, b_{j'}, t_p), (a_k, b_{j'}, t_p)$ because $I_\theta(a_k, b_{j'})$ is nested in $I_\theta(a_i, b_{j'})$. If $t_p$ lies in $I_\theta(a_k, b_{j'})$ then we are done. If not, because of the constraint on the position of cameras $t_p$ can lie only to the left of $I_\theta(a_k, b_{j'})$. Since $j' < n = j$, $I_\theta(a_i, b_j)$ will be nested in $I_\theta(a_i, b_{j'})$. Also $l$ cannot be greater than $j'$ because if it is the case $I_\theta(a_k, b_l)$ will be nested in $I_\theta(a_k, b_{j'})$ and since $t_1$ is the leftmost camera $t_p$ will lie to the right of it and hence will be covered by $I_\theta(a_k, b_l)$ as well, implying that it will be covered by $I_\theta(a_i, b_{j'})$ as well which is contrary to our assumption. Therefore, $l < j'$. Figure 3 shows the relative position of the intervals in this case. Hence in we can use the pairings $(a_i, b_j, t_1), (a_k, b_l, t_p), (a_{i'}, b_{j'}, t_q)$ to get a valid new solution. □

Therefore we can state the following theorem

**Theorem 5** *Under the constraint $C$, the Parallel-MAX-MIN problem can be solved in $O(n^3 \log^2 n)$ time.*

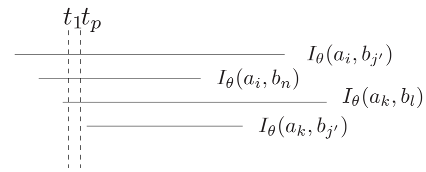


Figure 3: Case 3

**Lemma 6** *If all the targets are placed to the left of the right end point of the interval $I_\theta(a_n, b_n)$ and the instance has a solution then there exists a valid assignment in which the leftmost camera is assigned to the leftmost target.*

**Proof.** Consider the leftmost target $t_1$, two possible cases may arise. If it lies between the left end point of the interval $I_\theta(a_1, b_1)$ and the left end point of the interval $I_\theta(a_2, b_1)$, then the leftmost camera has to be used for it and in that case the lemma will be true. If it is not the case, then let in the real assignment it be covered by the interval $I_\theta(a_k, b_l)$, where $l \neq n$. Let in the real assignment the triples be $(a_k, b_l, t_1), (a_1, b_{j'}, t_p), (a_{i'}, b_n, t_q)$. Since $k < n$ and $I_\theta(a_1, b_n)$ also covers $t_1$ we can get a new solution by swapping, with new triples as $(a_1, b_n, t_1), (a_k, b_{j'}, t_p), (a_{i'}, b_l, t_q)$. Using similar arguments as in case 3 of the proof of correctness of the previous algorithm we can interchange in this case as well. Now when we have $l = n$, let the pairings in the solution be $(a_k, b_n, t_1), (a_1, b_x, t_p)$ (refer to Figure 4). Again using similar arguments as in previous case we can show that $(a_1, b_n, t_1), (a_k, b_x, t_p)$ is a valid interchange. Therefore in all the possible cases we can conclude that the left extreme camera is used with the left extreme target. Hence we get the desired result. □

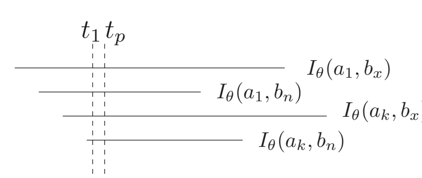


Figure 4: When $l = n$

Using the above lemma we can prove a result for the case when targets can be placed arbitrarily on the line. For the following lemma assume that the given instance of the decision problem has a valid assignment.

**Theorem 7** *When the cameras are positioned such that the interval property is satisfied then there exists a valid* *assignment in which the leftmost camera is assigned to the leftmost target or the rightmost camera is assigned to the rightmost target.*

**Proof.** We prove the above theorem using induction.
**Basis** For $n = 2$ the above statement can easily be verified by case analysis.
**Induction Step** Let the statement be true for the all values smaller than $n$. Now, when there are $n$ cameras let us assume that both the extreme cameras are not assigned to the respective extreme targets. Let the triples $(a_i, b_j, t_1)$ and $(a_k, b_l, t_n)$ belong to the solution where $i \neq 1$ and $l \neq 1$. Remove the triple $(a_k, b_l, t_n)$, the new setting with $2(n-1)$ cameras and $n-1$ targets also satisfies the interval property and has a solution. So, we can apply the induction hypothesis for this set of cameras and targets. If the leftmost camera is assigned to the leftmost target then we are done, else the rightmost target in the new setting, which is $t_{n-1}$, is paired with the rightmost camera. Let the triple corresponding to $t_{n-1}$ be $(a_{i'}, b_1, t_{n-1})$. Now put the triple $(a_k, b_l, t_n)$ back, to attain the original setting. Two cases arise, either $i' < k$ or $i' > k$. If $i' > k$ then it is not difficult to see that both the targets must lie in the intersection of the two intervals $I_\theta(a_k, b_l)$ and $I_\theta(a_{i'}, b_1)$ because of the relative position of the intervals (refer to Figure 5). So we can swap the assignments as $(a_{i'}, b_1, t_n), (a_k, b_l, t_{n-1})$ which implies that the rightmost camera is assigned to the rightmost target. Now, if $i' < k$ then the interval $I_\theta(a_k, b_l)$ is nested in $I_\theta(a_{i'}, b_1)$. Now $t_{n-1}$ cannot lie between the right endpoint of $I_\theta(a_k, b_l)$ and right end point of $I_\theta(a_{i'}, b_1)$, because it is to the left of $t_n$. So the only possible case when we cannot swap is that when $t_{n-1}$ lies between the left end-point of $I_\theta(a_{i'}, b_1)$ and the left endpoint of $I_\theta(a_l, b_l)$. So, in that case we remove the triples $(a_k, b_l, t_n)$ and $(a_{i'}, b_1, t_{n-1})$ from the instance to have a set of $n-2$ targets and $2(n-2)$ cameras which satisfy the interval property and has a solution. Since all the targets of the new set of cameras and targets lie to the left of $t_{n-1}$, they all lie to the left of the right end point of the smallest possible interval for the new set as well. And from lemma 6 there exists a solution in which the leftmost camera is paired with the leftmost target. Hence proved. □

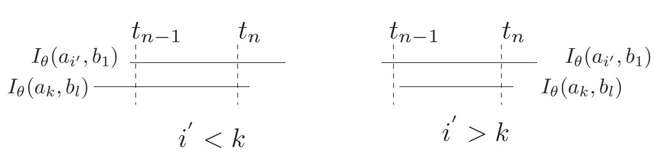


Figure 5: The two cases for $i'$

## 4 Conclusion

We study the Parallel-MAX-MIN version of the Focus Of Attention Problem and give a polynomial time algorithm assuming a constraint on the positioning of the cameras and the targets.

## References

[1] Volkan Isler, Sanjeev Khanna, John Spletzer and Camillo J. Taylor, *Target tracking with distributed sensors: The focus of attention problem*, Computer Vision and Image Understanding 100, pp. 225-247, 2005.

[2] Beat Gfeller,Matúš Mihal'ák, Subhash Suri, Elias Vicari and Peter Widmayer *Angle Optimization in Target Tracking*, 11th Scandinavian Workshop on Algorithm Theory (SWAT), pp. 65-76, 2008

[3] A. Yilmaz, O. Javed and M. Shah *Object tracking: A survey*, ACM Computing Surveys 38, 2006

[4] R.I. Hartley and A. Zisserman *Multiple View Geometry in Computer Vision*, Second edition, Cambridge University Press, ISBN: 0521540518, 2004