Bilattice-based Logical Reasoning for Human Detection[*]

Vinay D. Shet[†] Jan Neumann Visvanathan Ramesh
Siemens Corporate Research,
755 College Rd East,
Princeton, NJ
{vinay.shet; jan.neumann; visvanathan.ramesh}@siemens.com

Larry S. Davis
Computer Vision Laboratory,
University of Maryland,
College Park, MD
lsd@cs.umd.edu

Abstract

The capacity to robustly detect humans in video is a critical component of automated visual surveillance systems. This paper describes a bilattice based logical reasoning approach that exploits contextual information and knowledge about interactions between humans, and augments it with the output of different low level detectors for human detection. Detections from low level parts-based detectors are treated as logical facts and used to reason explicitly about the presence or absence of humans in the scene. Positive and negative information from different sources, as well as uncertainties from detections and logical rules, are integrated within the bilattice framework. This approach also generates proofs or justifications for each hypothesis it proposes. These justifications (or lack thereof) are further employed by the system to explain and validate, or reject potential hypotheses. This allows the system to explicitly reason about complex interactions between humans and handle occlusions. These proofs are also available to the end user as an explanation of why the system thinks a particular hypothesis is actually a human. We employ a boosted cascade of gradient histograms based detector to detect individual body parts. We have applied this framework to analyze the presence of humans in static images from different datasets.

1. Introduction

The primary objective of an automated visual surveillance system is to observe and understand human behavior and report unusual or potentially dangerous activities/events in a timely manner. Realization of this objective requires at its most basic level the capacity to robustly detect humans from input video. Human detection, however, is a difficult problem. This difficulty arises due to wide variability in appearance of clothing, articulation, view point changes, illumination conditions, shadows and reflections, among other factors. While detectors can be trained to handle some of these variations and detect humans individually as a whole, their performance degrades when humans are only partially visible due to occlusion, either by static structures in the scene or by other humans. Part based detectors are better suited to handle such situations because they can be used to detect the un-occluded parts. However, the process of going from a set of partial body part detections to a set of scene consistent, context sensitive, human hypotheses is far from trivial.

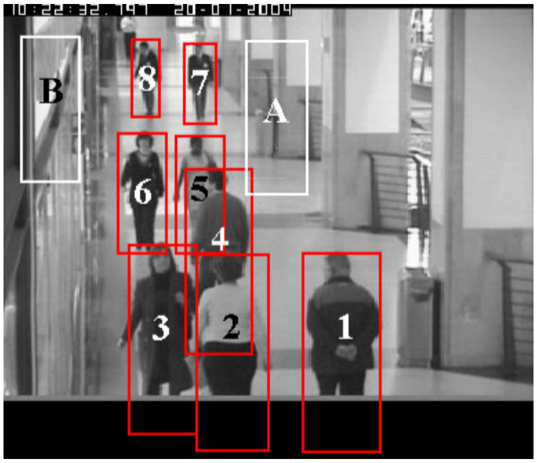

Figure 1. Figure showing valid human detections and a few false positives.

Since part based detectors only learn part of the information from the whole human body, they are typically less reliable and tend to generate large numbers of false positives. Occlusions and local image noise characteristics also lead to missed detections. It is therefore important to not only exploit contextual, scene geometry and human body constraints to weed out false positives, but also be able to explain as many valid missing body parts as possible to correctly detect occluded humans.

Figure 1 shows a number of humans that are occluded by the scene boundary as well as by each other. Ideally, a human detection system should be able to reason about whether a hypothesis is a human or not by aggregating information provided by different sources, both visual and non-visual. For example, in figure 1, the system should reason that it is likely that individual 1 is human because two independent sources, the head detector and the torso detector report that it is a human. The absence of legs indicates it is possibly not a human, however this absence can

[*]This research was funded in part by the U.S. Government's VACE program. [†]This paper was written when the first author was affiliated with the Computer Vision Laboratory at the University of Maryland

1-4244-1180-7/07/$25.00 ©2007 IEEE

be justified due to their occlusion by the image boundary. Furthermore, hypothesis 1 is consistent with the scene geometry and lies on the ground plane. Since the evidence for it being human exceeds evidence against, the system should decide that it is indeed a human. Similar reasoning applies to individual 4, only its legs are occluded by human 2. Evidence against A and B (inconsistent with scene geometry and not on the ground plane respectively) exceeds evidence in favor of them being human and therefore A and B should be rejected as being valid hypotheses.

This paper proposes a logic based approach that reasons and detects humans in the manner outlined above. In this framework, knowledge about contextual cues, scene geometry and human body constraints is encoded in the form of rules in a logic programming language and applied to the output of low level parts based detectors. Positive and negative information from different rules, as well as uncertainties from detections are integrated within the bilattice framework. This framework also generates proofs or justifications for each hypothesis it proposes. These justifications (or lack thereof) are further employed by the system to explain and validate, or reject potential hypotheses. This allows the system to explicitly reason about complex interactions between humans and handle occlusions. These proofs are also available to the end user as an explanation of why the system thinks a particular hypothesis is actually a human. We employ a boosted cascade of gradient histograms based detector to detect individual body parts.

We have applied this framework to analyze the presence of humans in static images and have evaluated it on the 'USC pedestrian set B' [22], USC's subset of the CAVIAR dataset [1], that includes images of partially occluded humans (This dataset will henceforth be referred to in this paper as the USC-CAVIAR dataset). We have also evaluated it on a dataset we collected on our own. In this paper, we refer to this dataset as Dataset-A.

2. Related Work

Approaches to detect humans from images/video tend to fall primarily in two categories: those that detect the human as a whole and those that detect humans based on part detectors. Among approaches that detect humans as a whole, Leibe et.al [11] employs an iterative method combining local and global cues via a probabilistic segmentation, Gavrilla [8, 7] uses edge templates to recognize full body patterns, Papageorgiou et. al. [15] uses SVM detectors, and Felzenszwalb [4] uses shape models. A popular detector used in such systems is a cascade of detectors trained using AdaBoost as proposed by Viola and Jones [20]. Such an approach uses as features several haar wavelets and has been very successfully applied for face detection in [20]. In [21] Viola and Jones applied this detector to detect pedestrians and made an observation that haar wavelets are insufficient by themselves as features for human detection and augmented their system with simple motion cues to get better performance. Another feature that is increasing in popularity is the histogram of oriented gradients. It was introduced by Dalal and Triggs [3] who used a SVM based classifier. This was further extended by Zhu et. al [24] to detect whole humans using a cascade of histograms of oriented gradients.

Part based representations have also been used to detect humans. Wu and Nevatia [22] use edgelet features and learn nested cascade detectors [10] for each of several body parts and detect the whole human using an iterative probabilistic formulation. Mikolajczyk et al. [12] divides the human body into seven parts and for each part a Viola-Jones approach is applied to orientation features. Mohan et.al. [13] divides the human into four different parts and learns SVM detectors using Haar wavelet features. [23, 22, 11] follow up low level detections with some form of high level reasoning that allows them to enforce global constraints, weed out false positives, and increase accuracy.

Logical reasoning has been used in visual surveillance applications to recognize the occurrence of different human activities [17] and, in conjunction with the bilattice framework, to maintain and reason about human identities as well [18].

3. Reasoning Framework

Logic programming systems employ two kinds of formulae, facts and rules, to perform logical inference. Rules are of the form "$A \leftarrow A_0, A_1, \cdots, A_m$" where each A_i is called an atom and ',' represents logical conjunction. Each atom is of the form $p(t_1, t_2, \cdots, t_n)$, where t_i is a term, and p is a predicate symbol of arity n. Terms could either be variables (denoted by upper case alphabets) or constant symbols (denoted by lower case alphabets). The left hand side of the rule is referred to as the head and the right hand side is the body. Rules are interpreted as "if body then head". Facts are logical rules of the form "$A \longleftarrow$" (henceforth denoted by just "A") and correspond to the input to the inference process. Finally, '$\neg$' represents negation such that $A = \neg\neg A$. In visual surveillance, rules typically capture knowledge about the proposition to be reasoned about and facts are the output of the low level computer vision algorithms onto which the rules are applied [18, 17].

3.1. Logic based Reasoning

To perform the kind of reasoning outlined in section 1, one has to specify rules that allow the system to take visual input from the low level detectors and explicitly infer whether or not there exists a human at a particular location. For instance, if we were to employ a head, torso and legs detector, then a possible rule would be:

$$\begin{aligned} human(X,Y,S) \quad \longleftarrow \quad & head(X_h,Y_h,S_h), \\ & torso(X_t,Y_t,S_t), \\ & legs(X_l,Y_l,S_l), \\ geometry\_constraint(&X_h,Y_h,S_h,X_t,Y_t,S_t,X_l,Y_l,S_l), \\ compute\_center(&X_h,Y_h,S_h,X_t,Y_t,S_t,X_l,Y_l,S_l,X,Y,S). \end{aligned}$$

This rule captures the information that if the head, torso and legs detectors were to independently report a detection at some location and scale (by asserting facts $head(X_h,Y_h,S_h)$, $torso(X_t,Y_t,S_t)$, $legs(X_l,Y_l,S_l)$ respectively), and these coordinates respected certain geometric constraints, then one could conclude that there exists a human at that location and scale. A logic programming system would search the input facts to find all combinations that satisfy the rule and report the presence of humans at those locations. Note that this rule will only detect humans that are visible in their entirety. Similar rules can be specified for situations when one or more of the detections are missing due to occlusions or other reasons. There are, however, some problems with a system built on such rule specifications:

1. Traditional logics treat such rules as binary and definite, meaning that every time the body of the rule is true, the head will have to be true. For a real world system, we need to be able to assign some uncertainty values to the rules that capture its reliability.

2. Traditional logics treat facts as binary. We would like to take as input, along with the detection, the uncertainty of the detection and integrate it into the reasoning framework

3. Traditional logic programming has no support for explicit negation in the head. There is no easy way of specifying a rule like:

$$\neg human(X, Y, S) \leftarrow \neg scene\_consistent(X, Y, S).$$

and integrating it with positive evidence. Such a rule says a hypothesis is not human if it is inconsistent with scene geometry.

4. Such a system will not be scalable. We would have to specify one rule for every situation we foresee. If we would like to include in our reasoning the output from another detector, say a hair detector to detect the presence of hair and consequently a head, we would have to re-engineer all our rules to account for new situations. We would like a framework that allows us to directly include new information without much re-engineering.

5. Finally, traditional logic programming does not have support for integration of evidence from multiple sources.

3.2. Bilattice Theory

Bilattices are algebraic structures introduced by Ginsberg [9] as a uniform framework within which a number of diverse applications in artificial intelligence can be modelled. In [9] Ginsberg used the bilattice formalism to model first order logic, assumption based truth maintenance systems, and formal systems such as default logics and circumscription. In [2], it was pointed out that bilattices serve as a foundation of many areas such as logic programming, computational linguistics, distributed knowledge processing, reasoning with imprecise information and fuzzy set theory. In our application, the automatic human detection system is looked upon as a passive rational agent capable of reasoning under uncertainty. Uncertainties assigned to the rules that guide reasoning, as well as detection uncertainties reported by the low level detectors, are taken from a set structured as a bilattice. These uncertainty measures are ordered along two axes, one along the source's[1] degree of information and the other along the agent's degree of belief. As we will see, this structure allows us to address all of the issues raised in the previous section and provides a uniform framework which not only permits us to encode multiple rules for the same proposition, but also allows inference in the presence of contradictory information from different sources.

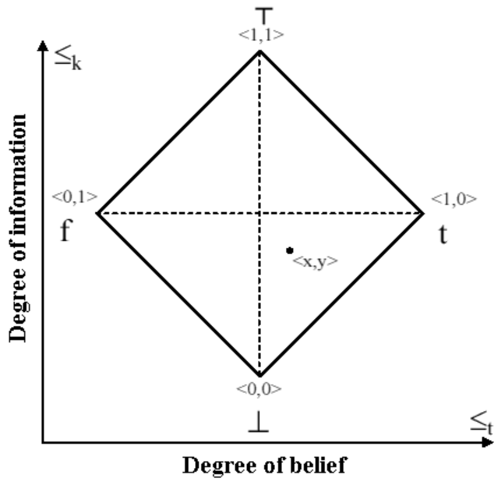

Figure 2. The bilattice square $([0, 1]^2, \leq_t, \leq_k)$

Definition 1 (Lattice) *A lattice is a set L equipped with a partial ordering $\leq$ over its elements, a greatest lower bound (glb) and a lowest upper bound (lub) and is denoted as $\mathcal{L} = (L, \leq)$ where glb and lub are operations from $L \times L \rightarrow L$ that are idempotent, commutative and associative. Such a lattice is said to be* ***complete****, iff for every nonempty subset M of L, there exists a unique lub and glb.*

Definition 2 (Bilattice [9]) *A bilattice is a triple $\mathcal{B} = (B, \leq_t, \leq_k)$, where B is a nonempty set containing at least two elements and $(B, \leq_t)$, $(B, \leq_k)$ are complete lattices.*

Informally a bilattice is a set, B, of uncertainty measures composed of two complete lattices $(B, \leq_t)$ and $(B, \leq_k)$ each of which is associated with a partial order $\leq_t$ and $\leq_k$ respectively. The $\leq_t$ partial order (agent's degree of belief) indicates how true or false a particular value is, with f being the minimal and t being the maximal while the $\leq_k$ partial order indicates how much is known about a particular proposition. The minimal element here is $\perp$ (completely unknown) while the maximal element is $\top$ (representing a contradictory state of knowledge where a proposition is both true and false). The glb and the lub operators on the $\leq_t$ partial order are $\wedge$ and $\vee$ and correspond to the usual logical notions of conjunction and disjunction, respectively. The glb and the lub operators on the $\leq_k$ partial order are $\otimes$ and $\oplus$, respectively, where $\oplus$ corresponds to the combination of evidence from different sources or lines of reasoning while $\otimes$ corresponds to the consensus operator. A bilattice is also equipped with a negation operator $\neg$ that inverts the sense of the $\leq_t$ partial order while leaving the $\leq_k$ partial order intact and a conflation operator $-$ which inverts the sense of the $\leq_k$ partial order while leaving the $\leq_t$ partial order intact.

The intuition is that every piece of knowledge, be it a rule or an observation from the real world, provides different degrees of information. An agent that has to reason about the state of the world based on this input, will have to translate the source's degree of information, to its own degree of belief. Ideally, the more information a source provides, the more strongly an agent is likely to believe it (i.e closer to the extremities of the t-axis) . The only exception to this rule being the case of contradictory information. When two

[1] A single rule applied to a set of facts is referred to as a source here. There can be multiple rules deriving the same proposition (both positive and negative forms of it) and therefore we have multiple sources of information.

sources contradict each other, it will cause the agent's degree of belief to decrease despite the increase in information content. It is this decoupling of the sources and the ability of the agent to reason independently along the truth axis that helps us address the issues raised in the previous section. It is important to note that the line joining $\bot$ and $\top$ represents the line of indifference. If the final uncertainty value associated with a hypothesis lies along this line, it means that the `degree of belief for` and `degree of belief against` it cancel each other out and the agent cannot say whether the hypothesis is true or false. Ideally the final uncertainty values should be either f or t, but noise in observation as well as less than completely reliable rules ensure that this is almost never the case. The horizontal line joining t and f is the line of consistency. For any point along this line, the `degree of belief for` will be exactly equal to (`1-degree of belief against`) and thus the final answer will be exactly consistent.

Definition 3 (Rectangular Bilattice [5, 14]) *Let $\mathcal{L} = (L, \leq_L)$ and $\mathcal{R} = (R, \leq_R)$ be two complete lattices. A rectangular bilattice is a structure $\mathcal{L} \odot \mathcal{R} = (L \times R, \leq_t, \leq_k)$, where for every $x_1, x_2 \in \mathcal{L}$ and $y_1, y_2 \in \mathcal{R}$,*

1. $\langle x_1, y_1 \rangle \leq_t \langle x_2, y_2 \rangle \Leftrightarrow x_1 \leq_L x_2$ *and* $y_1 \geq_R y_2$,
2. $\langle x_1, y_1 \rangle \leq_k \langle x_2, y_2 \rangle \Leftrightarrow x_1 \leq_L x_2$ *and* $y_1 \leq_R y_2$

An element $\langle x_1, y_1 \rangle$ of the rectangular bilattice $\mathcal{L} \odot \mathcal{R}$ may be interpreted such that x_1 represents the amount of belief for some assertion while y_1 represents the amount of belief against it. If we denote the glb and lub operations of complete lattices $\mathcal{L} = (L, \leq_L)$, and $\mathcal{R} = (R, \leq_R)$ by $\wedge_L$ and $\vee_L$, and $\wedge_R$ and $\vee_R$ respectively, we can define the glb and lub operations along each axis of the bilattice $\mathcal{L} \odot \mathcal{R}$ as follows:

$$\begin{aligned}
\langle x_1, y_1 \rangle \wedge \langle x_2, y_2 \rangle &= \langle x_1 \wedge_L x_2, y_1 \vee_R y_2 \rangle, \\
\langle x_1, y_1 \rangle \vee \langle x_2, y_2 \rangle &= \langle x_1 \vee_L x_2, y_1 \wedge_R y_2 \rangle, \\
\langle x_1, y_1 \rangle \otimes \langle x_2, y_2 \rangle &= \langle x_1 \wedge_L x_2, y_1 \wedge_R y_2 \rangle, \\
\langle x_1, y_1 \rangle \oplus \langle x_2, y_2 \rangle &= \langle x_1 \vee_L x_2, y_1 \vee_R y_2 \rangle \qquad (1)
\end{aligned}$$

Of interest to us in our application is a particular class of rectangular bilattices where $\mathcal{L}$ and $\mathcal{R}$ coincide. These structures are called *squares* [2] and $\mathcal{L} \odot \mathcal{L}$ is abbreviated as $\mathcal{L}^2$. Since detection likelihoods reported by the low level detectors are typically normalized to lie in the [0,1] interval, the underlying lattice that we are interested in is $\mathcal{L} = ([0,1], \leq)$. The bilattice that is formed by $\mathcal{L}^2$ is depicted in figure 2. Each element in this bilattice is a tuple with the first element encoding evidence for a proposition and the second encoding evidence against. In this bilattice, the element f (false) is denoted by the element $\langle 0, 1 \rangle$ indicating, no evidence for but full evidence against, similarly element t is denoted by $\langle 1, 0 \rangle$, element $\bot$ by $\langle 0, 0 \rangle$ indicating no information at all and $\top$ is denoted by $\langle 1, 1 \rangle$. To fully define glb and lub operators along both the axes of the bilattice as listed in equations 1, we need to define the glb and lub operators for the underlying lattice $([0,1], \leq)$. A popular choice for such operators are triangular-norms and triangular-conorms. Triangular norms and conorms were introduced by Schweizer and Sklar [16] to model the distances in probabilistic metric spaces. Triangular norms are used to model the glb operator and the triangular conorm to model the lub operator within each lattice.

Definition 4 (triangular norm) *A mapping*
$$\mathcal{T} : [0,1] \times [0,1] \rightarrow [0,1]$$
is a triangular norm (t-norm) iff $\mathcal{T}$ satisfies the following properties:
- Symmetry: $\mathcal{T}(a,b) = \mathcal{T}(b,a), \forall a, b \in [0,1]$
- Associativity: $\mathcal{T}(a, \mathcal{T}(b,c)) = \mathcal{T}(\mathcal{T}(a,b),c), \forall a,b,c \in [0,1]$.
- Monotonicity:$\mathcal{T}(a,b) \leq \mathcal{T}(a',b') if a \leq a' and b \leq b'$
- One identity: $\mathcal{T}(a,1) = a, \forall a \in [0,1]$.

Definition 5 (triangular conorm) *A mapping*
$$\mathcal{S} : [0,1] \times [0,1] \rightarrow [0,1]$$
is a triangular conorm (t-conorm) iff $\mathcal{S}$ satisfies the following properties:
- Symmetry: $\mathcal{S}(a,b) = \mathcal{S}(b,a), \forall a, b \in [0,1]$
- Associativity: $\mathcal{S}(a, \mathcal{S}(b,c)) = \mathcal{S}(\mathcal{S}(a,b),c), \forall a,b,c \in [0,1]$.
- Monotonicity:$\mathcal{S}(a,b) \leq \mathcal{S}(a',b') if a \leq a' and b \leq b'$
- Zero identity: $\mathcal{S}(a,0) = a, \forall a \in [0,1]$.

if $\mathcal{T}$ is a t-norm, then the equality $\mathcal{S}(a,b) = 1 - \mathcal{T}(1-a, 1-b)$ defines a t-conorm and we say $\mathcal{S}$ is derived from $\mathcal{T}$. There are number of possible t-norms and t-conorms one can choose. In our application, for the underlying lattice, $\mathcal{L} = ([0,1], \leq)$, we choose the t-norm such that $\mathcal{T}(a,b) \equiv a \wedge_L b = ab$ and consequently choose the t-conorm as $\mathcal{S}(a,b) \equiv a \vee_L b = a + b - ab$. Based on this, the glb and lub operators for each axis of the bilattice B can then be defined as per equation 1.

3.3. Inference

Inference in bilattice based reasoning frameworks is performed by computing the closure over the truth assignment.

Definition 6 (Truth Assignment) *Given a declarative language L, a truth assignment is a function $\phi : L \rightarrow B$ where B is a bilattice on truth values or uncertainty measures.*

Definition 7 (Closure) *Let $\mathcal{K}$ be the knowledge base and ϕ be a truth assignment labelling each every formula $k \in \mathcal{K}$. The closure over ϕ, denoted $cl(\phi)$, is the truth assignment that labels information that is entailed by $\mathcal{K}$.*

For example, if ϕ labels sentences $\{p, (q \leftarrow p)\} \in \mathcal{K}$ as $\langle 1, 0 \rangle$ (true); i.e. $\phi(p) = \langle 1, 0 \rangle$ and $\phi(q \leftarrow p) = \langle 1, 0 \rangle$, then $cl(\phi)$ should also label q as $\langle 1, 0 \rangle$ as it is information entailed by $\mathcal{K}$. Entailment is denoted by the symbol '$\models$' ($\mathcal{K} \models q$).

Denote by S a set of sentences entailing q. The uncertainty measure to be assigned to the conjunction of elements of S should be

$$\bigwedge_{p \in S} cl(\phi)(p) \qquad (2)$$

This term represents the conjunction of the closure of the elements of S^2. It is important to note that this term is

[2] Recall that $\wedge$ and $\vee$ are glb and lub operators along the $\leq_t$ ordering and $\otimes$ and $\oplus$ along $\leq_k$ axis. $\bigwedge, \bigvee, \bigotimes, \bigoplus$ are their infinitary counterparts such that $\bigoplus_{p \in S} p = p_1 \oplus p_2 \oplus \cdots$ and so on

Assume the following set of rules and facts:

Rules	Facts
$\phi(human(X,Y,S) \leftarrow head(X,Y,S)) = \langle 0.40, 0.60\rangle$	$\phi(head(25,95,0.9)) = \langle 0.90, 0.10\rangle$
$\phi(human(X,Y,S) \leftarrow torso(X,Y,S)) = \langle 0.30, 0.70\rangle$	$\phi(torso(25,95,0.9)) = \langle 0.70, 0.30\rangle$
$\phi(\neg human(X,Y,S) \leftarrow \neg scene\_consistent(X,Y,S)) = \langle 0.90, 0.10\rangle$	$\phi(\neg scene\_consistent(25,95,0.9)) = \langle 0.80, 0.20\rangle$

Inference is performed as follows:

$$cl(\phi)(human(25,95,0.9)) = \langle 0,0\rangle \vee \Big[\langle 0.4,0.6\rangle \wedge \langle 0.9,0.1\rangle\Big] \oplus \langle 0,0\rangle \vee \Big[\langle 0.3,0.7\rangle \wedge \langle 0.7,0.3\rangle\Big] \oplus \neg\Bigg(\langle 0,0\rangle \vee \Big[\langle 0.9,0.1\rangle \wedge \langle 0.8,0.2\rangle\Big]\Bigg)$$

$$= \langle 0.36,0\rangle \oplus \langle 0.21,0\rangle \oplus \neg\langle 0.72,0\rangle = \langle 0.4944,0\rangle \oplus \langle 0,0.72\rangle = \langle 0.4944,0.72\rangle$$

Figure 3. Example showing inference using closure within a $([0,1]^2, \leq_t, \leq_k)$ bilattice

not the final uncertainty value to be assigned to q, rather it is merely a contribution to its final value. The reason it is merely a contribution is because there could be other sets of sentences S' that entail q, representing different lines of reasoning (or, in our case, different rules). These contributions need to be combined using the $\oplus$ operator along the information ($\leq_k$) axis. Also, if the expression in 2 evaluates to false, then its contribution to the value of q should be $\langle 0,0\rangle$ (unknown) and not $\langle 0,1\rangle$ (false). These arguments suggest that the closure over ϕ of q is

$$cl(\phi)(q) = \bigoplus_{S\models q} \perp \vee [\bigwedge_{p\in S} cl(\phi)(p)] \tag{3}$$

where $\perp$ is $\langle 0,0\rangle$. We also need to take into account the set of sentences entailing $\neg q$. Aggregating this information yields the following expression:

$$cl(\phi)(q) = \bigoplus_{S\models q} \perp \vee [\bigwedge_{p\in S} cl(\phi)(p)] \oplus \neg \bigoplus_{S\models \neg q} \perp \vee [\bigwedge_{p\in S} cl(\phi)(p)] \tag{4}$$

For more details see [9]. Figure 3 shows an example illustrating the process of computing the closure as defined above by combining evidence from three sources. In this example, the final uncertainty value computed is $\langle 0.4944, 0.72\rangle$. This indicates that evidence against the hypothesis at (25,95) at scale 0.9 exceeds evidence in favor of and, depending on the final threshold for detection, this hypothesis is likely to be rejected.

3.4. Negation

Systems such as this typically employ different kinds of negation. One kind of negation that has already been mentioned earlier is $\neg$. This negation flips the bilattice along the $\leq_t$ axis while leaving the ordering along the $\leq_k$ axis unchanged. Another important kind of negation is negation by failure to prove, denoted by not. $not(A)$ succeeds if A fails. This operator flips the bilattice along both the $\leq_t$ axis as well as the $\leq_k$ axis. Recall that, in section 3, $-$ was defined as the conflation operator that flips the bilattice along the $\leq_k$ axis. Therefore, $\phi(not(A)) = \neg - \phi(A)$. In other words, if A evaluates to $\langle 0,0\rangle$, then $not(A)$ will evaluate to $\langle 1,1\rangle$. This operator is important when we want to detect the absence of a particular body part for a hypothesis.

4. Detection System

Rules can now be defined within this bilattice framework to handle complex situations, such as humans being partially occluded by static structures in the scene or by other humans. Each time one of the detectors detects a body part, it asserts a logical fact of the form $\phi(head(x,y,s)) = \langle \alpha, \beta\rangle$, where α is the measurement score the detector returns at that location and scale in the image and, for simple detectors, β is $1-\alpha$. Rules are specified similarly as $\phi(human(X,Y,S) \leftarrow \cdots) = \langle \gamma, \delta\rangle$. γ and δ are learnt as outlined in subsection 4.2. We start by initializing a number of initial hypotheses based on the low level detections. For example, if the head detector detects a head and asserts fact $\phi(head(75,225,1.25)) = \langle 0.95, 0.05\rangle$[3], the system records that there exists a possible hypothesis at location (75,225) at scale 1.25 and submits the query $human(75,225,1.25)$ to the logic program where support for and against it is gathered and finally combined into a single answer within the bilattice framework. Projecting the final uncertainty value onto the $\langle 0,1\rangle - \langle 1,0\rangle$ axis, gives us the final degree of belief in the hypothesis. We will now provide English descriptions of some of the rules employed in our system.

4.1. Rule Specification

Rules in such systems can be learnt automatically; however, such approaches are typically computationally very expensive. We manually encode the rules while automatically learning the uncertainties associated with them. The rules fall into three categories: Detector based, Geometry based and Explanation based

Detector based: These are the simplest rules that hypothesize that a human is present at a particular location if one or more of the detectors detects a body part there. In other words, if a head is detected at some location, we say there exists a human there. There are positive rules, one each for the head, torso, legs and fullbody based detectors as well as negative rules that fire in the absence of these detections.

Geometry based: Geometry based rules validate or reject human hypotheses based on geometric and scene information. This information is entered a priori in the system at setup time. We employ information about expected height of people and regions of expected foot location. The ex-

[3] Note that the coordinates here are not the centers of the body parts, but rather the centers of the body

pected image height rule is based on ground plane information and anthropometry. Fixing a gaussian at an adult human's expected physical height allows us to generate scene consistency likelihoods for a particular hypothesis given its location and size. The expected foot location region is a region demarcated in the image outside of which no valid feet can occur and therefore serves to eliminate false positives.

Explanation based: Explanation based rules are the most important rules for a system that has to handle occlusions. The idea here is that if the system does not detect a particular body part, then it must be able to explain its absence for the hypothesis to be considered valid. If it fails to explain a missing body part, then it is construed as evidence against the hypothesis being a human. Absence of body parts is detected using logic programming's 'negation as failure' operator (not). $not(A)$ succeeds when A evaluates to $\langle 0, 0\rangle$ as described in section 3.4. A valid explanation for missing body part could either be due to occlusions by static objects or due to occlusions by other humans.

Explaining missed detections due to occlusions by static objects is straightforward. At setup, all static occlusions are marked. Image boundaries are also treated as occlusions and marked as shown in figure 1(black area at bottom of figure). For a given hypothesis, the fraction of overlap of the missing body part with the static occlusion is computed and reported as the uncertainty of occlusion. The process is similar for occlusions by other human hypotheses, with the only difference being that, in addition to the degree of occlusion, we also take into account the degree of confidence of the hypothesis that is responsible for the occlusion, as illustrated in the rule below:

$$\begin{aligned} human(X,Y,S) \quad \leftarrow \quad & not(torso(X_t,Y_t,S_t), \\ & torso\_body\_consistent(X,Y,S,X_t,Y_t,S_t)), \\ & torso\_occluded(X,Y,S,X_o,Y_o,S_o), \\ & Y_o > Y, human(X_o,Y_o,S_o). \end{aligned} \tag{5}$$

This rule will check to see if $human(X,Y,S)$'s torso is occluded by $human(X_o,Y_o,S_o)$ under condition that $Y_o > Y$, meaning the occluded human is behind the 'occluder'[4] There is a similar rule for legs and also rules deriving $\neg human$ in the absence of explanations for missing parts.

4.2. Learning

Given a rule of the form $A \leftarrow B_1, B_2, \cdots, B_n$, a confidence value of $\langle \mathcal{F}(A|B_1,B_2,\cdots,B_n), \mathcal{F}(\neg A|B_1,B_2,\cdots,B_n)\rangle$ is computed, where $\mathcal{F}(A|B_1,B_2,\cdots,B_n)$ is the fraction of times A is true when $B_1, B_2, \cdots, B_n$ is true.

4.3. Generating Proofs

As mentioned earlier, in addition to using the explanatory ability of logical rules, we can also provide these explanations to the user as justification of why the system believes that a given hypothesis is a human. The system provides a straightforward technique to generate proofs from its inference tree. Since all of the bilattice based reasoning is encoded as meta-logical rules in a logic programming language, it is easy to add predicates that succeed when the rule fires and propagate character strings through the inference tree up to the root where they are aggregated and displayed. Such proofs can either be dumps of the logic program itself or be English text. In our implementation, we output the logic program as the proof tree.

5. Body Part Detector

Our human body part detectors are inspired by [24]. Similar to their approach we train a cascade of svm-classifiers on histograms of gradient orientations. Instead of the hard threshold function suggested in their paper, we apply a sigmoid function to the output of each svm. These softly thresholded functions are combined using a boosting algorithm [6]. After each boosting round, we calibrate the probability of the partial classifier based on evaluation set, and set cascade decision thresholds based on the sequential likelihood ratio test similar to [19]. To train the parts-based detector, we restrict the location of the windows used during the feature computation to the areas corresponding to the different body parts (head/shoulder, torso, legs).

6. Experiments

The framework has been implemented in C++ with an embedded Prolog reasoning engine. The C++ module initializes the Prolog engine by inserting into its knowledge base all predefined rules. Information about scene geometry, and static occlusions is specified through the user interface, converted to logical facts and inserted into the knowledge base. The C++ module then runs the detectors on the given image, clusters the detector output, and finally structures the clustered output as logical facts for the Prolog knowledge base. Initial hypotheses are created based on these facts and then evidence for or against these hypotheses is searched for by querying for them. We will first describe some qualitative results and show how our system reasons and resolves difficult scenarios, and then describe quantitative results on the USC-CAVIAR dataset as well as on Dataset-A.

6.1. Qualitative Results

Tables 1 and 2 list the proofs for humans 1 and 4 from figure 1. In both cases, the head and torso are visible while the legs are missing. In case of human 1, it is due to occlusion by the image boundary (which has been marked as a static occlusion) and in case of human 4 due to occlusion by human 2. In tables 1 and 2, variables starting with $\_G\cdots$ are non-unified variables in Prolog, meaning that legs cannot be found and therefore the variables of the predicate legs cannot be instantiated. It can be seen that in both cases, evidence in favor of the hypothesis exceeds that against.

6.2. Numerical Results

We applied our framework to the set of static images taken from USC-CAVIAR dataset. This dataset, a subset of

[4] The reader might notice that calling the $human(X_o,Y_o,S_o)$ within the definition of a 'human' rule will cause the system to infer the presence of $human(X_o,Y_o,S_o)$ from scratch. This rule has been presented in such a manner merely for ease of explication. In practice, we maintain a table of inferences that the query, $human(X_o,Y_o,S_o)$, can tap into for unification without re-deriving anything. Also we derive everything from the bottom of the image to the top, so $human(X_o,Y_o,S_o)$, if it exists, is guaranteed to unify.

Total:	human(243,253,1.5)	$\langle 0.484055, 0.162474\rangle$
+ve evidence:	head(244.5, 247.5, 1.5)	$\langle 1, 0\rangle$
	torso(243, 253,1.5)	$\langle 1, 0\rangle$
	fullbody(243, 256.5,1.5)	$\langle 0.9371, 0.0629\rangle$
	on_ground_plane(243, 253, 1.5),	$\langle 1, 0\rangle$
	scene_consistent(243, 253, 1.5),	$\langle 0.954835, 0.045165\rangle$
	not((legs(_G3817, _G3818,_G3819),	
	legs_body_consistent(243, 253, 1.5, _G3817,_G3818, _G3819)))	$\langle 1, 1\rangle$
	is_part_occluded(219.0, 253.0, 267.0, 325.0)	$\langle 0.569444, 0.430556\rangle$
-ve evidence:	¬ scene_consistent(243, 253, 1.5)	$\langle 0.045165, 0.954835\rangle$
	not((legs(_G3984,_G3985, _G3986),	
	legs_body_consistent(243, 253, 1.5, _G3984,_G3985, _G3986)))	$\langle 1, 1\rangle$

Table 1. Proof for human marked as ‘1’ in figure 1

Total:	human(154,177,1.25)	$\langle 0.359727, 0.103261\rangle$
+ve evidence:	head(154, 177, 1.25)	$\langle 0.94481, 0.05519\rangle$
	torso(156.25, 178.75, 1.25)	$\langle 0.97871, 0.02129\rangle$
	on_ground_plane(154, 177, 1.25)	$\langle 1, 0\rangle$
	scene_consistent(154, 177, 1.25)	$\langle 0.999339, 0.000661\rangle$
	not((legs(_G7093,_G7094, _G7095),	
	legs_body_consistent(154, 177, 1.25,_G7093, _G7094, _G7095)))	$\langle 1, 1\rangle$
	is_part_occluded(134.0, 177.0, 174.0, 237.0)	$\langle 0.260579, 0.739421\rangle$
-ve evidence:	¬scene_consistent(154, 177, 1.25)	$\langle 0.000661, 0.999339\rangle$
	not((legs(_G7260, _G7261, _G7262),	
	legs_body_consistent(154, 177, 1.25,_G7260, _G7261, _G7262)))	$\langle 1, 1\rangle$

Table 2. Proof for human marked as ‘4’ in figure 1

the original CAVIAR [1] data, contains 54 frames with 271 humans of which 75 humans are partially occluded by other humans and 18 humans are occluded by the scene boundary. This data is not part of our training set. We have trained our parts based detector on the MIT pedestrian dataset [15]. For training purposes, the size of the human was 32x96 centered and embedded within an image of size 64x128. We used 924 positive images and 6384 negative images for training. The number of layers used in fullbody, head, torso and leg detectors were 12, 20, 20, and 7 respectively. Figure 4 shows the ROC curves for our parts based detectors as well as for the full reasoning system. “Full Reasoning*”, in Figure 4, is the ROC curve on the 75 occluded humans and table 3 lists detection rates for these 75 humans for different degrees of occlusion. ROC curves for part based detectors represent detections that have no prior knowledge about scene geometry or other anthropometric constraints. It can be seen that performing high level reasoning over low level part based detections, especially in presence of occlusions, greatly increases overall performance. We have also compared the performance of our system with the results reported by Wu and Nevatia [22] on the same dataset. We have taken results reported in their original paper and plotted them in figure 4 as well as listed them in table 3. As can be seen, results from both systems are comparable.

Occlusion Degree(%)	>70	70-50	50-25
Human#	10	31	34
Detection Rate(%) (interpolated to 19 false alarms)	87	91.4	92.6
Detection Rate(%) (Wu Nevatia [22])	80	90.3	91.2

Table 3. Detection rates on the USC-CAVIAR dataset for different degrees of occlusion on the 75 humans that are occluded by other humans (with 19 false alarms). Results of [22] on the same dataset are copied from their original paper.

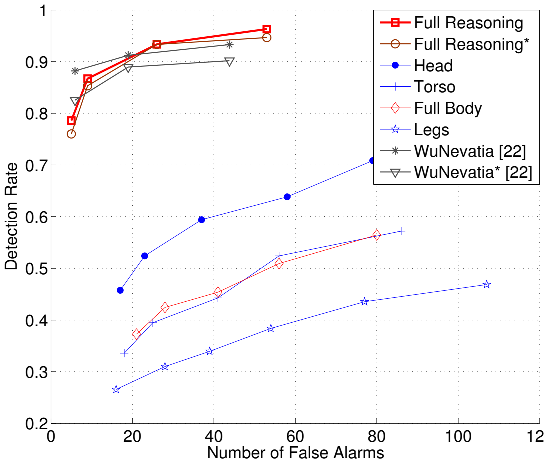

Figure 4. ROC curves for evaluation on the USC-CAVIAR dataset. Full Reasoning* is ROC curve for 75 humans occluded by other humans. Results of [22] on the same dataset are copied from their original paper. WuNevatia* is ROC curve for the 75 humans occluded by other humans

We also applied our framework on another set of images taken from a dataset we collected on our own (in this paper we refer to it as Dataset-A). This dataset contains 58 images (see figure 5) of 166 humans, walking along a corridor, 126 of whom are occluded 30% or more, 64 by the image boundary and 62 by each other. Dataset-A is significantly harder than the USC-CAVIAR dataset due to heavier occlusions (44 humans are occluded 70% or more), perspec-

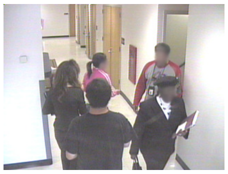

Figure 5. An image from Dataset-A

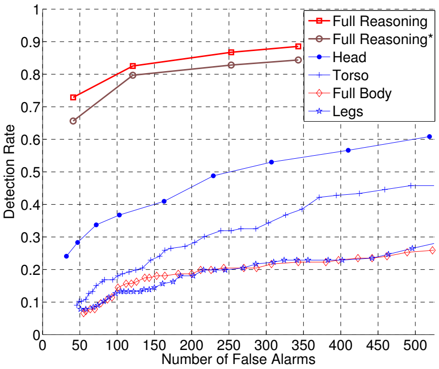

Figure 6. ROC curves for evaluation on Dataset-A. Full Reasoning* is ROC curve for 126 occluded humans.

tive distortions (causing humans to appear tilted), and due to the fact that many humans appear in profile view. Figure 6 shows the ROC curves for this dataset. It can be seen that the low level detectors as well as the full body detector perform worse here than on the USC-CAVIAR data, however, even in such a case, the proposed logical reasoning approach gives a big improvement in performance. If the performance of the low level detectors is further enhanced (to take in account profile views and handle perspective distortions), then results of high level reasoning will further improve. This is part of our future work.

7. Discussions and Future Work

We have described a logical reasoning approach for human detection that takes input from multiple sources of information, both visual and non-visual, and integrates them into a single hypothesis within the bilattice framework. Use of logical reasoning permits to explicitly reason about complex interactions between humans as well as with the environment and thus handle occlusions. Structuring of this reasoning within the bilattice framework makes it scalable, so information from new sources can be added easily and also allows use of explicitly negative information about a hypothesis, providing for a better separation between true positives and false alarms. The system also generates proofs for validation by the operator. Finally, as can be seen from the closure expression (equation 4), complexity of inference in such systems is linear in the number of rules and its constituent propositions. In the future we would like to extend this system to reason explicitly about temporal information thus helping us not only track humans, but also to define models for and recognize human activities within a single framework.

References

[1] CAVIAR homepage:http://homepages.inf.ed.ac.uk/rbf/caviar/.

[2] O. Arieli, C. Cornelis, G. Deschrijver, and E. Kerre. Bilattice-based squares and triangles. *Lecture Notes in Computer Science: Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 563–575, 2005.

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR05*, pages I: 886–893, 2005.

[4] P. Felzenszwalb. Learning models for object recognition. In *CVPR01*, pages I:1056–1062, 2001.

[5] M. C. Fitting. Bilattices in logic programming. In *20th International Symposium on Multiple-Valued Logic, Charlotte*, pages 238–247. IEEE CS Press, Los Alamitos, 1990.

[6] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journ. of Comp. and System Sciences*, 55:119–139, 1997.

[7] D. Gavrila. Pedestrian detection from a moving vehicle. In *ECCV00*, pages II: 37–49, 2000.

[8] D. Gavrila and V. Philomin. Real-time object detection for smart vehicles. In *ICCV99*, pages 87–93, 1999.

[9] M. L. Ginsberg. Multivalued logics: A uniform approach to inference in artificial intelligence. *Computational Intelligence*, 4(3):256–316, 1988.

[10] C. Huang, H. Al, B. Wu, and S. Lao. Boosting nested cascade detector for multi-view face detection. In *ICPR04*, pages II: 415–418, 2004.

[11] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *IEEE CVPR'05 in , San Diego, CA*, pages 878–885. sp, may 2005.

[12] K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *ECCV*, May 2004.

[13] A. Mohan, C. Papageorgiou, and T. Poggio. Example-based object detection in images by components. *PAMI*, 23(4):349–361, April 2001.

[14] O.Arieli, C.Cornelis, and G.Deschrijver. Preference modeling by rectangular bilattices. *Proc. 3rd International Conference on Modeling Decisions for Artificial Intelligence (MDAI'06)*, (3885):22–33, April 2006.

[15] C. Papageorgiou, T. Evgeniou, and T. Poggio. A trainable pedestrian detection system. *Intelligent Vehicles*, pages 241–246, October 1998.

[16] B. Schweizer and A. Sklar. Associative functions and abstract semigroups. *Publ. Math. Debrecen*, 1963.

[17] V. Shet, D. Harwood, and L. Davis. Vidmap: video monitoring of activity with prolog. In *IEEE AVSS*, pages 224–229, 2005.

[18] V. Shet, D. Harwood, and L. Davis. Multivalued default logic for identity maintenance in visual surveillance. In *ECCV*, pages IV: 119–132, 2006.

[19] J. Sochman and J. Matas. Waldboost – learning for time constrained sequential detection. CVPR 2005.

[20] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. IEEE Conference on Computer Vision and Pattern Recognition (CVPR'01), 2001.

[21] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *ICCV03*, pages 734–741, 2003.

[22] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. *ICCV*, Oct 2005. Beijing.

[23] T. Zhao and R. Nevatia. Bayesian human segmentation in crowded situations. *CVPR*, 2:459–466, 2003.

[24] Q. Zhu, M. Yeh, K. Cheng, and S. Avidan. Fast human detection using a cascade of histograms of oriented gradients. In *CVPR06*, pages II: 1491–1498, 2006.