

# Разработка средств сбора и логического анализа 3D-видеоданных на основе времяпролётной камеры и Акторного Пролога

А. А. Морозов<sup>a,b</sup>, О. С. Сушкова<sup>a</sup>, А. Ф. Полупанов<sup>a</sup>

<sup>a</sup>Институт радиотехники и электроники им. В. А. Котельникова РАН, 125009, ул. Моховая, 11-7, Москва, Россия

<sup>b</sup>Московский городской психолого-педагогический университет, 107045, ул. Ретенка 29, Москва, Россия

---

## Аннотация

Предложен подход к интеллектуальному 3D-видеонаблюдению на основе объектно-ориентированного логического программирования. В отличие от обычного 2D-видеонаблюдения, методы трёхмерного зрения обеспечивают надёжное распознавание частей тела, что делает возможным новые постановки задач и практическое применение методов анализа поведения людей в системах видеонаблюдения. Логический подход к интеллектуальному видеонаблюдению позволяет описывать сложное поведение людей на основе определений простых действий и поз. Цель данной работы заключается в реализации этих преимуществ логического подхода в области интеллектуального 3D-видеонаблюдения.

*Ключевые слова:* интеллектуальное видеонаблюдение; трёхмерное зрение; объектно-ориентированное логическое программирование; Акторный Пролог; Кинект; скелетоны; распознавание сложных событий; машинное зрение; техническое зрение

---

## 1. Введение

Термин «трёхмерное видеонаблюдение» используется разными исследователями по-разному:

1. Первоначально этот термин использовался в отношении систем видеонаблюдения, осуществляющих в реальном времени трёхмерную визуализацию контролируемого пространства, а именно, создающих 3D-модели зданий, сооружений и наблюдаемых людей. Для трёхмерной визуализации при этом могут использоваться технологии виртуальной реальности и дополненной реальности [1, 2, 3, 4].
2. Этот термин можно использовать в отношении так называемых «трёхмерных» подходов к оценке движения объектов на основе их двумерных изображений [5].
3. В настоящее время термин «трёхмерное видеонаблюдение» (3D-видеонаблюдение) можно связать с применением в области видеонаблюдения различного оборудования для ввода в компьютер трёхмерной видеоинформации (трёх- и двас-половиной-мерных данных, 3D и 2.5D) [6]. Примерами такого оборудования являются времяпролётные (time-of-flight, ToF) камеры, камеры на основе структурированного света, стереокамеры, лазерные локаторы (LiDARs) [7], импульсные лазерные локаторы (Flash Ladars) [8]. Далее, для простоты, мы будем называть все эти приборы оборудованием для ввода трёхмерных данных.

Появление оборудования для ввода трёхмерных данных существенным образом ускорило развитие различных областей компьютерного зрения, связанных с интеллектуальным видеонаблюдением, таких как вычитание фона и трекинг людей [9, 10], идентификация наблюдаемых людей [11, 12], распознавание лиц [13], распознавание поз [14, 15], распознавание жестов [16], подсчёт людей [17], распознавание действий людей [18] и др. Это связано, прежде всего, с тем, что использование трёхмерных данных открыло новые возможности для решения некоторых труднопреодолимых проблем в области компьютерного зрения, связанных с наличием теней, изменением освещения, изменением положения камеры, непредсказуемым изменением фона, перекрытием изображений объектов и пр. Однако гораздо более важным является то, что эти технологические достижения стали предпосылками для решения качественно более сложных проблем интеллектуального видеонаблюдения [19], таких как распознавание и предсказание взаимодействия между людьми [20], выявление необычных событий [21, 22, 23, 24], исследование поведения покупателей [25], видеонаблюдение за инвалидами и людьми с психическими расстройствами [26, 27], распознавание агрессивного поведения и раздражённого состояния человека [28], распознавание людей по походке [29], биометрическая идентификация [30], анализ движений человека для оценки его клинического состояния и выбора лечебных процедур [31, 32].

В отличие от обычного «двумерного» видеонаблюдения, методы трёхмерного компьютерного зрения могут обеспечить надёжное распознавание частей тела человека, и это делает возможным постановку новых задач и практическое применение методов анализа поведения людей в системах интеллектуального видеонаблюдения. Согласно классификации [33], «поведение человека» (behavior) является более сложным понятием, чем просто «активность человека» (activity). Поведением называется активность человека, являющаяся откликом на некоторые стимулы (внешние, внутренние, осознанные или неосознанные) [33]. Например, разновидностями поведения являются социальные виды взаимодействия людей, такие как дружеское рукопожатие, драка или кража чемодана у другого человека. Это означает, что для анализа поведения людей необходимо учитывать информацию о контексте наблюдаемых событий и об иных факторах, влияющих на поведение. Такая информация может включать геометрические параметры видеосцены, сведения о наличии (отсутствии) и местоположении каких-то предметов, временные ограничения и т. п. Таким образом, возможность получить с помощью средств трёхмерного зрения точную информацию о положении различных частей тела человека и геометрических параметрах сцены является ключевым элементом в решении задач анализа, распознавания и понимания поведения человека.

Общепринятым подходом к описанию и анализу контекстной информации о видеосцене является применение средств математической логики и логического программирования. Логическое программирование – область исследований на стыке математической логики и теории программирования; применение логического программирования для анализа поведения людей означает, что для описания и анализа активности людей используются логические правила (формулы). При этом логические правила используются также и для описания знаний о форме и свойствах человеческого тела, свойствах различных предметов, временных ограничениях, геометрических параметрах видеосцены и пр. Для описания этих правил используется некоторый логический язык со строго определённой математической семантикой, а исходными данными для логического вывода на основе этих правил являются выходные данные некоторых низкоуровневых процедур, осуществляющих детектирование людей и предметов.

Идея использования методов и средств логического программирования для реализации систем интеллектуального видеонаблюдения была предложена и использована в ряде исследовательских проектов в рамках обычного двумерного видеонаблюдения. Одной из первых экспериментальных систем для анализа действий и перемещений людей, в которой были применены средства временного логического программирования, является система  $W^4$  [34]. Логический язык Пролог был применён в экспериментальной системе интеллектуального видеонаблюдения VidMAP [35]. В архитектуре этой системы предусмотрено сочетание средств логического программирования с низкоуровневыми алгоритмами обработки видео в реальном времени. Позднее эта же исследовательская группа разработала специальное расширение логики предикатов для нечёткого логического вывода по данным видеонаблюдения [36]. В логику предикатов был добавлен математический аппарат бирешёток. Ещё одной исследовательской системой интеллектуального видеонаблюдения, основанной на использовании языка Пролог, является VERSA [37]. Математический аппарат Answer Set Programming, также широко применяемый в современном логическом программировании, был использован в работе [38] для обнаружения в реальном времени сложных аудиовизуальных событий. В работе [39] для описания эвристических правил анализа взаимодействия людей была применена временная логика на основе интервальной алгебры. В работе [40] описана система распознавания так называемых «длительных» событий, состоящих из последовательностей «краткосрочных» событий. Примерами длительных событий являются «драка» и «встреча людей». В состав длительных событий могут входить краткосрочные события «ходьба», «бег», «активное движение на месте», «неподвижное состояние», а также так называемые «резкие движения». Для анализа поведения людей применён аппарат исчисления событий, реализованный стандартными средствами логического программирования. Позднее этот подход был расширен и дополнен возможностью учитывать степень неопределённости результатов распознавания активности человека [41]. Для этого был применён язык вероятностного логического программирования ProbLog. В работе [42] описана система интеллектуального видеонаблюдения, основанная на языке Пролог, предназначенная для надёжного обнаружения оставленных без присмотра предметов. Система автоматически выявляет взаимосвязь между наблюдаемыми людьми для того чтобы сделать вывод о том, кому именно принадлежат наблюдаемые предметы, и тем самым уменьшить вероятность ложной тревоги.

Интересно, что до сих пор вышло лишь небольшое число работ, в которых логическое программирование применяется для анализа трёхмерных видеоданных. В частности, в работе [26] метод распознавания длительных событий [40] адаптирован для анализа данных, получаемых с помощью сенсора Kinect. Разработанная система мониторинга предназначена для анализа поведения пациентов с психическими проблемами. По сообщению [26], для анализа длительных событий не используется информация о структуре тела (скелетоны), однако при этом вычисляется некий коэффициент уровня активности пациента на основе усреднённых расстояний перемещения всех узлов скелетона (joints), координаты которых были непосредственно измерены (статус узла «tracked») или выведены (статус узла «inferred») сенсором Кинект. В работе [43] для создания распределённой системы мониторинга был адаптирован вероятностный подход к распознаванию длительных событий [41]. Система предназначена для мониторинга состояния человека в рамках концепции умного дома. В работе [44] данные с сенсора Кинект используются для распознавания поз и жестов. Для анализа трёхмерных данных применён механизм логического вывода на основе дескриптивной логики. Правила в стиле языка Пролог применяются в экспериментальной системе RoboSherlock [45] для различения действий «взять», «положить», «наполнить», «толкать». Логический вывод осуществляется с учётом контекстной информации о свойствах предметов. Например, для распознавания действия «наполнить», необходимо знать о том, что предмет, которым манипулирует человек, является ёмкостью или пищевым продуктом.

Логический подход к реализации интеллектуального видеонаблюдения обладает следующими очевидными достоинствами [26]:

1. Информация о контексте наблюдаемых событий описывается и анализируется простым и естественным образом с помощью логических формул.
2. Ход рассуждений о поведении людей может быть очень сложным, однако для разработчика системы интеллектуального видеонаблюдения он всегда остаётся принципиально понимаемым.
3. Логический подход к интеллектуальному видеонаблюдению позволяет описывать сложное поведение людей на основе определений простых действий и поз.

Целью данной работы является реализация указанных преимуществ логического подхода в области интеллектуального трёхмерного видеонаблюдения.

## **2. Объектно-ориентированное логическое программирование систем интеллектуального трёхмерного видеонаблюдения**

Метод объектно-ориентированного логического программирования систем интеллектуального видеонаблюдения был разработан авторами в рамках парадигмы обычного двумерного видеонаблюдения [46, 47, 48, 49, 50, 51, 52]. Отличительными особенностями данного метода являются использование объектно-ориентированного логического языка Акторный Пролог [53, 54, 55] и трансляция логических программ интеллектуального видеонаблюдения в язык Джава [47, 56]. В Акторном Прологе текст логической программы состоит из классов [53], а параллельные процессы являются разновидностью

экземпляров классов [54]. Объектно-ориентированные средства языка позволяют разделять логическую программу на взаимодействующие между собой параллельные процессы, реализующие различные стадии обработки видеопотока и анализа видеосцены. При этом трансляция в язык Джава обеспечивает скорость обработки данных, достаточную для анализа видео в реальном времени, а также надёжность и устойчивость работы программ интеллектуального видеонаблюдения.

Принято различать этапы низкоуровневой и высокоуровневой обработки видеопотока в процессе анализа поведения человека. В случае традиционного двумерного видеонаблюдения, этап низкоуровневой обработки видео обычно включает вычитание фона, детектирование на видеоизображении людей и транспортных средств, вычисление траекторий перемещений движущихся людей и объектов, оценку их скорости и пр. Этап высокоуровневого анализа может включать логический анализ графов траекторий движения людей и объектов, вычисленных алгоритмами низкоуровневого анализа, а также визуализацию результатов анализа [50, 51]. Принципиальным отличием этапов низкоуровневой и высокоуровневой обработки видеопотока является то, что для достижения наилучшей производительности системы видеонаблюдения они реализованы на языках программирования разного уровня. В частности, в Акторном Прологе низкоуровневые алгоритмы обработки видео реализованы на языке Джава [57] в виде специального встроенного класса.

В общем случае, средства обычного двумерного видеонаблюдения принципиально не могут обеспечить стабильное распознавание частей человеческого тела. Это относится, прежде всего, к практическим приложениям интеллектуального видеонаблюдения, характеризующимся плохими или непредсказуемыми условиями видеосъёмки. Поэтому в рамках нашего подхода к интеллектуальному 2D видеонаблюдению движение людей и объектов описывается с помощью графов траекторий блобов<sup>1</sup>. Граф траекторий блобов может содержать информацию о координатах и скорости блобов, а также набор статистических метрик, характеризующих плавность движения блобов на видеоизображении [48]. Метрики движения блобов вычисляются на этапе низкоуровневого анализа видеопотока и используются для различения на видеоизображении бегущих людей, велосипедистов и автомобилей. Как правило, для этого применяются простые процедуры нечёткого логического вывода, реализованные с помощью стандартных арифметических предикатов.

В случае трёхмерного видеонаблюдения, задачи, решаемые на этапе низкоуровневого анализа видеопотока, являются существенно иными. Например, стандартное программное обеспечение устройства Kinect 2, в составе которого использована времяпролётная видеокамера, вычисляет в реальном времени блобы и скелетоны наблюдаемых людей (до шести одновременно) [6, 58, 59, 18, 19, 60, 61]. Это означает, что этап низкоуровневого анализа видеоданных может состоять всего лишь из преобразования координат скелетонов в структуры данных логического языка. Если это необходимо, могут быть вычислены скорость и ускорение (первая и вторая производные координат) узлов. Высокоуровневый анализ видеоданных при этом может быть реализован на тех же принципах, что и в случае двумерного видеонаблюдения. Однако при этом доступная для логического анализа информация о видеосцене является существенно более полной, и это позволяет получить значительно большее число содержательных логических выводов о наблюдаемых событиях.

### 3. Пример логического вывода на основе трёхмерных видеоданных

Рассмотрим пример логической программы, анализирующей трёхмерные данные, собираемые с помощью времяпролётной камеры устройства Kinect 2.

Данные, передаваемые из устройства, после обработки алгоритмами стандартного SDK включают карты глубины, а также скелетоны людей (координаты узлов скелетонов и пр.). Рассмотренная в этом разделе логическая программа предназначена только для анализа скелетонов, карты глубины при этом будут использоваться лишь для визуализации трёхмерных данных. Каждый скелетон включает 25 узлов, которые могут находиться в разных состояниях (см. Рис. 1). Различаются три возможных состояния узла скелетона:

1. Tracked («прослеженный»), то есть, непосредственно наблюдаемый времяпролётной камерой.
2. Inferred («выведенный»), то есть, узел, положение которого является гипотетическим и получено на основании анализа координат других узлов.
3. Unknown («неизвестный»), то есть, узел, координаты которого определить не удалось.

Используя синтаксис Акторного Пролога, соответствующие скелетонам структуры данных могут быть описаны следующим образом:

```
DOMAINS:  
Skeletons      = Skeleton*.
```

Тип данных Skeletons является списком элементов данных типа Skeleton.

```
Skeleton = {  
    identifier: INTEGER,  
    head: SkeletonJoint,  
    spine: SkeletonSpine,  
    left_arm: SkeletonArm,  
    right_arm: SkeletonArm,  
    left_leg: SkeletonLeg,  
    right_leg: SkeletonLeg  
}.
```

<sup>1</sup>Блоб – выделенная область изображения переднего плана.

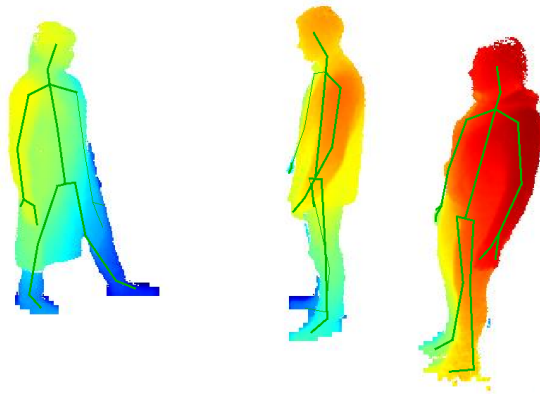


Рис. 1: Входными данными логической программы являются координаты скелетов и карта глубины. На рисунке карта глубины обозначена с помощью красно-синей палитры цветов. Скелетоны обозначены зелёными линиями. Узлы скелетона, находящиеся в состоянии «Tracked», соединены сплошными линиями; в иных случаях применены пунктирные линии.

Тип данных `Skeleton` является недоопределённым множеством [53] Акторного Пролога. Элементы множества соответствуют именам узлов скелетона, смысл которых понятен из их анатомических названий (шея, плечо, локоть, запястье, кисть руки, колено, лодыжка, ступня и пр.):

```
SkeletonSpine = {
    neck: SkeletonJoint,
    shoulder: SkeletonJoint,
    mid: SkeletonJoint,
    base: SkeletonJoint
}.

SkeletonArm = {
    shoulder: SkeletonJoint,
    elbow: SkeletonJoint,
    wrist: SkeletonJoint,
    hand: SkeletonJoint,
    tip: SkeletonJoint,
    thumb: SkeletonJoint
}.

SkeletonLeg = {
    hip: SkeletonJoint,
    knee: SkeletonJoint,
    ankle: SkeletonJoint,
    foot: SkeletonJoint
}.

SkeletonJoint = {
    status: SkeletonJointStatus,
    position: VertexPosition3D,
    orientation: JointOrientation3D
}.
```

Состояние узла обозначается с помощью символа *TRACKED*, *INFERRED* или *UNKNOWN*. Каждый узел характеризуется координатами в трёхмерном пространстве (то есть, тремя числами), ориентацией (описываемой четырьмя числами), а также двумя парами двумерных координат этого узла на инфракрасном и цветном изображениях, передаваемых из устройства Kinect 2. Будем считать также, что для каждого узла скелетона вычислены скорость и ускорение.

```
SkeletonJointStatus =
    'TRACKED'; 'INFERRED'; 'UNKNOWN'.

VertexPosition3D = {
    point: Point3D,
    velocity: Point3D,
    acceleration: Point3D,
    mapping1: Point2D,
    mapping2: Point2D
}.

JointOrientation3D= q(
    Numerical,
    Numerical,
    Numerical,
    Numerical).

Point3D = p(Numerical,Numerical,Numerical).
Point2D = p(Numerical,Numerical).
```

Определим набор предикатов, предназначенных для анализа трёхмерных данных:

```
PREDICATES:
imperative:
analyze_skeletons(Skeletons)          - (i);
nondeterm:
a_hand_of_a_person
    (Skeletons,INTEGER,VertexPosition3D) - (i,o,o);
determ:
is_near_a_head_of_a_colleague
    (Skeletons,INTEGER,VertexPosition3D) - (i,i,i);
is_near(VertexPosition3D,VertexPosition3D) - (i,i);
```

Предикат `analyze_skeletons` проверяет наличие в видеопотоке как минимум двух людей, один из которых наносит удар рукой в лицо другого человека. В случае обнаружения таких людей, на экран выводится сообщение (см. Рис. 2). Текст соответствующих логических правил приведён ниже:

```
CLAUSES:
analyze_skeletons(Skeletons):-
    a_hand_of_a_person(Skeletons,Id,Position),
    is_near_a_head_of_a_colleague(Skeletons,Id,Position),!,
    graphic_window ? draw_text(
        0.93,0.08,"A punch in the face is detected!").
analyze_skeletons(_).
```

**A punch in the face is detected!**



Рис. 2: Логическая программа обнаружила, что один человек ударил другого человека в лицо, и выдала сообщение «Зарегистрирован удар рукой в лицо!»

Предикат `a_hand_of_a_person` является недетерминированным. Этот предикат возвращает координаты рук людей, обнаруженных в видеопотоке<sup>2</sup>:

```
a_hand_of_a_person([Skeleton|_],Identifier,Position):-
    Skeleton == {
        identifier:Identifier,
        left_arm:{
            hand:{
                status:'TRACKED',
                position:Position|_}
            |_}
        |_}.
a_hand_of_a_person([Skeleton|_],Identifier,Position):-
    Skeleton == {
        identifier:Identifier,
        right_arm:{
            hand:{
                status:'TRACKED',
                position:Position|_}
            |_}
        |_}.
a_hand_of_a_person([_|Rest],Identifier,Position):-
    a_hand_of_a_person(Rest,Identifier,Position).
```

<sup>2</sup>В Акторном Прологе оператор `==` соответствует обычному оператору `=` (унификация аргументов) стандартного Пролога.

Исполнение предиката `is_near_a_head_of_a_colleague` заканчивается успехом в том случае, если переданные в него трёхмерные координаты руки одного человека близки к положению в пространстве головы другого человека. Для того чтобы это правило можно было использовать для распознавания агрессивных действий, предикат осуществляет дополнительную проверку того, что человек подносит руку к голове другого человека, а не к своей собственной. В целях упрощения изложения, скорость движения руки и взаимное положение людей в приведённом ниже правиле не учитываются, но в реальной программе интеллектуального видеонаблюдения эти факторы, конечно же, должны быть учтены.

```
is_near_a_head_of_a_colleague([Skeleton|_],Id1,Point1):-
    Skeleton == {
        identifier:Id2,
        head:{ status:'TRACKED',
              position:Point2|_}
    |_},
    Id1 <> Id2,
    is_near(Point1,Point2),!.
is_near_a_head_of_a_colleague([Skeleton|_],Id1,Point1):-
    Skeleton == {
        identifier:Id2,
        head:{ status:'TRACKED',
              position:Point2|_}
    |_},
    Id1 <> Id2,
    is_near(Point1,Point2),!.
is_near_a_head_of_a_colleague([_|Rest],Id,Point):-
    is_near_a_head_of_a_colleague(Rest,Id,Point).
```

Предикат `is_near` является вспомогательным. Он проверяет, что расстояние между указанными координатами в пространстве достаточно мало. Для этого осуществляются простейшие вычисления с помощью встроенных арифметических предикатов:

```
is_near({point:p(X1,Y1,Z1)|_},{point:p(X2,Y2,Z2)|_):-
    Delta== ?sqrt(
        ?power(X1-X2,2) +
        ?power(Y1-Y2,2) +
        ?power(Z1-Z2,2)),
    Delta < 0.15,!.

```

Чтобы упростить изложение, мы не будем рассматривать предикаты, осуществляющие ввод данных из устройства Kinect 2 и общую организацию логического анализа видеосцены. Пример определения аналогичных предикатов можно найти в работе [50]. Общая схема анализа заключается в том, что в логической программе предикат `analyze_skeletons` должен вызываться время от времени, чтобы проверять положение узлов скелетонов, обнаруженных в видеопотоке, и делать вывод о наличии или отсутствии искомого события.

Заметим, что смысл приведённого примера заключается вовсе не в распознавании удара рукой, что является довольно простой задачей, а в том, что аналогичным образом могут быть проверены более сложные логические формулы. Например, с помощью немного более сложных логических правил можно определить события «угроза оружием», «прицеливание в человека», «паническое поведение группы людей» и др. Не меньший интерес может представлять и распознавание простых повседневных событий, таких как рукопожатие. Например, можно проверить логическое условие, все ли люди поздоровались с вошедшим в комнату человеком или нет. Анализируя, кто из здоровающихся людей первым протянул руку, можно оценить, каков социальный статус здоровающихся людей. А в случае непрерывного видеонаблюдения, через некоторое время логическая программа может, в принципе, определить, каковы социальные отношения между всеми людьми, побывавшими в наблюдаемой комнате, и составить об этом подробную базу знаний.

#### 4. Заключение

Применение методов логического объектно-ориентированного программирования для анализа данных, получаемых с помощью оборудования трёхмерного зрения, открывает новые возможности в области интеллектуального видеонаблюдения. Дальнейшее развитие этой области требует междисциплинарных исследований и будет определяться достижениями в разработке новых средств ввода трёхмерных данных и алгоритмов низкоуровневого анализа карт глубины, промышленных языков и систем логического программирования общего назначения, методов логического анализа трёхмерных видеосцен.

#### Благодарности

Авторы благодарны Ивану А. Кершнеру и Ренате А. Толмачёвой, ИРЭ им. В. А. Котельникова РАН, за помощь в подготовке образцов трёхмерных видеозаписей, а также Angelos Varnpoutis, университет штата Флорида США, за предоставленную им библиотеку J4K [59], которую мы использовали для низкоуровневой обработки данных.

Работа выполнена при поддержке РФФИ, грант № 16-29-09626-офи\_м.

## Список литературы

- [1] 3D video surveillance with augmented virtual environments / I. O. Sebe, J. Hu, S. You, U. Neumann // First ACM SIGMM international workshop on Video surveillance / ACM. — 2003. — Pp. 107–112.
- [2] Advanced virtual reality technologies for surveillance and security applications / R. Ott, M. Gutiérrez, D. Thalmann, F. Vexo // International Conference on Virtual Reality Continuum and Its Applications / ACM. — 2006. — Pp. 163–170.
- [3] Dynamic seethroughs: Synthesizing hidden views of moving objects / P. Barnum, Y. Sheikh, A. Datta, T. Kanade // Mixed and Augmented Reality (ISMAR 2009) / IEEE. — 2009. — Pp. 111–114.
- [4] An immersive system for browsing and visualizing surveillance video / P. DeCamp, G. Shaw, R. Kubat, D. Roy // International Conference on Multimedia / ACM. — 2010. — Pp. 371–380.
- [5] Gavrilă D.M. The visual analysis of human movement: A survey // Computer Vision and Image Understanding. — 1999. — January. — Vol. 73, no. 1. — Pp. 82–98.
- [6] Chen L., Wei H., Ferryman J. A survey of human motion analysis using depth imagery // Pattern Recognition Letters. — 2013. — Vol. 34, no. 15. — Pp. 1995–2006.
- [7] Benedek C. 3D people surveillance on range data sequences of a rotating Lidar // Pattern Recognition Letters. — 2014. — Vol. 50. — Pp. 149–158.
- [8] Stettner R., Bailey H., Silverman S. Three dimensional Flash LADAR focal planes and time dependent imaging // International Journal of High Speed Electronics and Systems. — 2008. — Vol. 18, no. 02. — Pp. 401–406.
- [9] Cluster tracking with time-of-flight cameras / D. W. Hansen, M. S. Hansen, M. Kirschmeyer et al. // Computer Vision and Pattern Recognition Workshops (CVPRW'08) / IEEE. — 2008. — Pp. 1–6.
- [10] Silvestre D. — Video surveillance using a time-of-flight camera. — Master's thesis, Informatics and Mathematical Modelling, Technical University of Denmark, 2007.
- [11] Real-time appearance-based person re-identification over multiple Kinect™ cameras / R. Satta, F. Pala, G. Fumera, F. Roli // VISAPP. — 2013. — Pp. 407–410.
- [12] Sinha A., Chakravarty K., Bhowmick B. Person identification using skeleton information from Kinect // The Sixth International Conference on Advances in Computer-Human Interactions (ACHI 2013). — 2013. — Pp. 101–108.
- [13] Krishnan P., Naveen S. RGB-D face recognition system verification using Kinect and FRAV3D databases // Procedia Computer Science. — 2015. — Vol. 46. — Pp. 1653–1660.
- [14] Diraco G., Leone A., Siciliano P. Human posture recognition with a time-of-flight 3D sensor for in-home applications // Expert Systems with Applications. — 2013. — Vol. 40, no. 2. — Pp. 744–751.
- [15] Easy gesture recognition for Kinect / R. Ibañez, Á. Soria, A. Teyseyre, M. Campo // Advances in Engineering Software. — 2014. — Vol. 76. — Pp. 171–180.
- [16] Robust gesture recognition using Kinect: A comparison between DTW and HMM / J. Raheja, M. Minhas, D. Prashanth et al. // Optik-International Journal for Light and Electron Optics. — 2015. — Vol. 126, no. 11. — Pp. 1098–1104.
- [17] A Kinect-based people-flow counting system / C.-T. Hsieh, H.-C. Wang, Y.-K. Wu et al. // Intelligent Signal Processing and Communication Systems (ISPACS 2012). — IEEE, 2012. — Pp. 146–150.
- [18] Aggarwal J. K., Xia L. Human activity recognition from 3D data: A review // Pattern Recognition Letters. — 2014. — Vol. 48. — Pp. 70–80.
- [19] Lun R., Zhao W. A survey of applications and human motion recognition with Microsoft Kinect // International Journal of Pattern Recognition and Artificial Intelligence. — 2015. — March. — Vol. 29, no. 05.
- [20] Alazrai R., Mowafi Y., Lee C. Anatomical-plane-based representation for human–human interactions analysis // Pattern Recognition. — 2015. — Vol. 48, no. 8. — Pp. 2346–2363.
- [21] Fall detection from depth map video sequences / C. Rougier, E. Auvinet, J. Rousseau et al. // International Conference on Smart Homes and Health Telematics / Springer. — 2011. — Pp. 121–128.
- [22] Lee Y.-S., Chung W.-Y. Visual sensor based abnormal event detection with moving shadow removal in home healthcare applications // Sensors. — 2012. — Vol. 12, no. 1. — Pp. 573–584.
- [23] Mastorakis G., Makris D. Fall detection system using Kinect's infrared sensor // Journal of Real-Time Image Processing. — 2014. — Vol. 9, no. 4. — Pp. 635–646.
- [24] Wang C., Liu H. Unusual events detection based on multi-dictionary sparse representation using Kinect // International Conference on Image Processing / IEEE. — 2013. — Pp. 2968–2972.
- [25] Kinect sensing of shopping related actions / M. Popa, A. K. Koc, L. J. Rothkrantz et al. // International Joint Conference on Ambient Intelligence / Springer. — 2011. — Pp. 91–100.
- [26] Johanna M. — Recognizing activities with the Kinect. A logic-based approach for the support room. — Master's thesis, Radboud University Nijmegen, 2013.
- [27] Lau T. B., Ong A. C., Putra F. A. Non-invasive monitoring of people with disabilities via motion detection // International Journal of Signal Processing Systems. — 2014. — June. — Vol. 2, no. 1. — Pp. 37–41.
- [28] Patwardhan A., Knapp G. Aggressive actions and anger detection from multiple modalities using Kinect // arXiv preprint arXiv:1607.01076. — 2016.
- [29] Gait recognition with Kinect / J. Preis, M. Kessel, M. Werner, C. Linnhoff-Popien // International Workshop on Kinect in Pervasive Computing. — Newcastle, UK: 2012.
- [30] Savage R., Clarke N., Li F. Multimodal biometric surveillance using a Kinect sensor // Proceedings of the 12th annual security conference, Las Vegas, USA. — 2013.
- [31] Chaaraoui A. A., Padilla-Lo'pez J. R., Flo'rez-Revuelta F. Abnormal gait detection with RGB-D devices using joint motion history features // Automatic Face and Gesture Recognition (FG) / IEEE. — Vol. 7. — 2015. — Pp. 1–6.
- [32] Sensing behaviour using the Kinect: Identifying characteristic features of instability and poor performance during challenging balancing tasks / D. Leightley, M. H. Yap, B. M. Hewitt, J. S. McPhee // Measuring Behavior. — 2016.
- [33] Borges P. V. K., Conci N., Cavallaro A. Video-based human behavior understanding: A survey // IEEE Transactions on Circuits and Systems for Video Technology. — 2013. — Vol. 23. — Pp. 1993–2008.
- [34] Haritaoglu I., Harwood D., Davis L.S. W<sup>4</sup>: Who? When? Where? What? A real time system for detecting and tracking people // FG 1998. — Nara, Japan: 1998. — Pp. 222–227.
- [35] Shet V., Harwood D., Davis L. VidMAP: Video monitoring of activity with Prolog // AVSS 2005. — IEEE, 2005. — Pp. 224–229.
- [36] Predicate logic based image grammars for complex pattern recognition / V. Shet, M. Singh, C. Bahlmann et al. // International Journal of Computer Vision. — 2011. — June. — Vol. 93, no. 2. — Pp. 141–161.
- [37] O'Hara S. VERSA—video event recognition for surveillance applications. M.S. thesis. University of Nebraska at Omaha. — 2008.
- [38] Real time complex event detection for resource-limited multimedia sensor networks / F. Machot, K. Kyamakya, B. Dieber, B. Rinner // AMMCSS 2011. — 2011. — Pp. 468–473.
- [39] Lao W., Han J., de With P. H. N. Flexible human behavior analysis framework for video surveillance applications // International Journal of Digital Multimedia Broadcasting. — 2010. — Vol. 2010. — Article ID 920121.
- [40] Artikis A., Sergot M., Paliouras G. A logic programming approach to activity recognition // International Workshop on Events in Multimedia (EiMM 2010). — NY, USA: ACM, 2010. — Pp. 3–8.
- [41] A probabilistic logic programming event calculus / A. Skarlatidis, A. Artikis, J. Filippou, G. Paliouras // Theory and Practice of Logic Programming. — 2014. — 9. — Pp. 1–33.
- [42] Robust abandoned object detection integrating wide area visual surveillance and social context / J. Ferryman, D. Hogg, J. Sochman et al. // Pattern Recognition Letters. — 2013. — Vol. 34, no. 7. — Pp. 789–798.
- [43] Katzouris N., Artikis A., Paliouras G. Event recognition for unobtrusive assisted living // Hellenic Conference on Artificial Intelligence / Springer. — 2014. — Pp. 475–488.
- [44] Semantic matchmaking for Kinect-based posture and gesture recognition / M. Ruta, F. Scioscia, M. D. Summa et al. // International Journal of Semantic Computing. — 2014. — Vol. 8, no. 4. — Pp. 491–514.
- [45] Worch J.-H., Bálint-Benczedi F., Beetz M. Perception for everyday human robot interaction // KI - Künstliche Intelligenz. — 2016. — Vol. 30, no. 1. — Pp. 21–27.

- [46] Development of concurrent object-oriented logic programming platform for the intelligent monitoring of anomalous human activities / A. A. Morozov, A. Vaish, A. F. Polupanov et al. // *BIOSTEC 2014* / Ed. by G. Plantier, T. Schultz, A. Fred, H. Gamboa. — Vol. 511 of *CCIS*. — Springer, 2015. — Pp. 82–97.
- [47] Morozov A. A., Polupanov A. F. Intelligent visual surveillance logic programming: Implementation issues // *CICLOPS-WLPE 2014* / Ed. by T. Ströder, T. Swift. — *Aachener Informatik Berichte* no. AIB-2014-09. — RWTH Aachen University, 2014. — June. — Pp. 31–45. <http://aib.informatik.rwth-aachen.de/2014/2014-09.pdf>.
- [48] Morozov A. A., Polupanov A. F. Development of the logic programming approach to the intelligent monitoring of anomalous human behaviour // *OGRW2014* / Ed. by D. Paulus, C. Fuchs, D. Droege. — Koblenz: University of Koblenz-Landau, 2015. — 5. — Pp. 82–85. [https://kola.opus.hbz-nrw.de/files/915/OGRW\\_2014\\_Proceedings.pdf](https://kola.opus.hbz-nrw.de/files/915/OGRW_2014_Proceedings.pdf).
- [49] Morozov A. A., Sushkova O. S., Polupanov A. F. An approach to the intelligent monitoring of anomalous human behaviour based on the Actor Prolog object-oriented logic language // *RuleML 2015 DC and Challenge* / Ed. by N. Bassiliades, P. Fodor, A. Giurca et al. — Berlin: CEUR, 2015. <https://www.csw.inf.fu-berlin.de/ruleml2015-ceur>.
- [50] Morozov A. A. Development of a method for intelligent video monitoring of abnormal behavior of people based on parallel object-oriented logic programming // *Pattern Recognition and Image Analysis*. — 2015. — Vol. 25, no. 3. — Pp. 481–492. <http://www.cplire.ru/Lab144/pria481.pdf>.
- [51] . . . . . видеоизображений в реальном времени средствами языка Акторный Пролог // . . . . . — 2016. — Vol. 40, no. 6. — Pp. 947–957.
- [52] Morozov A. A., Sushkova O. S. The intelligent visual surveillance logic programming Web Site. — 2017. <http://www.fullvision.ru/monitoring>.
- [53] Morozov A. A. Actor Prolog: an object-oriented language with the classical declarative semantics // *IDL 1999* / Ed. by K. Sagonas, P. Tarau. — Paris, France: 1999. — September. — Pp. 39–53. <http://www.cplire.ru/Lab144/paris.pdf>.
- [54] Morozov A. A. Logic object-oriented model of asynchronous concurrent computations // *Pattern Recognition and Image Analysis*. — 2003. — Vol. 13, no. 4. — Pp. 640–649. <http://www.cplire.ru/Lab144/pria640.pdf>.
- [55] Morozov A. A. Operational approach to the modified reasoning, based on the concept of repeated proving and logical actors // *CICLOPS 2007* / Ed. by S. Abreu, V. S. Costa. — Porto, Portugal: 2007. — September. — Pp. 1–15. <http://www.cplire.ru/Lab144/ciclops07.pdf>.
- [56] Morozov A. A., Sushkova O. S., Polupanov A. F. A translator of Actor Prolog to Java // *RuleML 2015 DC and Challenge* / Ed. by N. Bassiliades, P. Fodor, A. Giurca et al. — Berlin: CEUR, 2015. <https://www.csw.inf.fu-berlin.de/ruleml2015-ceur>.
- [57] Morozov A. A. A GitHub repository containing source codes of Actor Prolog built-in classes. — 2016. <https://github.com/Morozov2012/actor-prolog-java-library>.
- [58] Enhanced computer vision with Microsoft Kinect sensor: A review / J. Han, L. Shao, D. Xu, J. Shotton // *IEEE Transactions on Cybernetics*. — 2013. — October. — Vol. 43, no. 5. — Pp. 1318–1334.
- [59] Barmpoutis A. Tensor body: Real-time reconstruction of the human body and avatar synthesis from RGB-D // *IEEE Transactions on Cybernetics*. — 2013. — Vol. 43, no. 5. — Pp. 1347–1356.
- [60] Space-time representation of people based on 3D skeletal data: A review / F. Han, B. Reily, W. Hoff, H. Zhang // *arXiv preprint arXiv:1601.01006v2 [cs.CV]*. — 2016.
- [61] Presti L. L., Cascia M. L. 3D skeleton-based human action classification: A survey // *Pattern Recognition*. — 2016. — Vol. 53. — Pp. 130–147.