|  |
| --- |
| Ученическа конференция .... |
| Разпознаване на човешки активности на база 3D реконструкция на скелета |
| **Автор: Любомир Янчев;**  **ЧНГ „Ерих Кестнер“, София;** [yanchev.lyubomir@gmail.com](mailto:yanchev.lyubomir@gmail.com) |
| **Ръководител: Светослав Колев;  University of Washington;**  [swetko@cs.washington.edu](mailto:swetko@cs.washington.edu) |
|  |

|  |
| --- |
|  |

Съдържание

[Абстракт 2](#_Toc334655228)

[Въведение 2](#_Toc334655229)

[Предишна работа 2](#_Toc334655230)

[Методи 3](#_Toc334655231)

[Ъгли и ротация на ставите 3](#_Toc334655232)

[Представяне на ъглите чрез двуизмерни проекции координатите 3](#_Toc334655233)

[Представяне на ротацията и позицията на ставите чрез геометрични кватерниони 3](#_Toc334655234)

[Заключение 4](#_Toc334655235)

[Селекция на най-информативни стави 4](#_Toc334655236)

[Сравняване на движения 5](#_Toc334655237)

[Dynamic time warping (DWT) 6](#_Toc334655238)

[Разпознаване на активности в реално време 7](#_Toc334655239)

[Използване на „прозорец” 7](#_Toc334655240)

[Използване на маркери за начало и край на движението 7](#_Toc334655241)

[Заключение 7](#_Toc334655242)

[Имплементация 8](#_Toc334655243)

[Използвани технологии 8](#_Toc334655244)

[Kinect SDK 8](#_Toc334655245)

[C# & .NET 8](#_Toc334655246)

[Възможни приложения 8](#_Toc334655247)

[Заключение 8](#_Toc334655248)

[Използвана литература 9](#_Toc334655249)

# Абстракт

Наскоро нашумялата тема за разпознаване на човешки действия се радва на все повече популярност. Целта на настоящата разработка е да представи качествено ново решение на посочения проблем.

В разработения алгоритъм са използвани многоизмерни статистически алгоритми, алгоритми за филтриране на данни, алгоритми за оценка и други. Триизмерната информация за скелета се извлича от Kinect сензор. Чрез извършването на малка модификация, алгоритъмът би могъл да се използва за разпознаване и на друг вид зависимости.

В допълнение е направено сравнение между разработения алгоритъм и традиционно използваните скрити модели на Марков.

# Въведение

Разпознаването на човешки действия остава една от най-значимите сфери на изследване за компютърно зрение. Разработването на алгоритми за целта е стимулирано от многото потенциални приложения на подобна технология – от автоматизирано видео наблюдение, през базирано на съдържанието видео, управление на компютри до вграждането му в „умни домове“.

В миналото екстракцията на точна информация за скелета от монокулярни камери бе труден за реализация и несигурен подход, а ефективни алгоритми използващи друг вид сензори, не работеха на нужното ниво. Специализираните системи за следене на предмети, базирани на поставяне на маркери върху повърхността на обектите, предоставят изключително точни данни, но за сметка на това са приложими единствено в студийна среда, което ограничава приложението им. Изключение правят някои системи, които главно се използват за мониторинг на пациенти и други. [1] Въпреки това, те често се оказват неудобни, поради необходимостта от слагане на физически сензори по тялото на потребителя.

Поради тази причина се налага обществото от учени да се средоточи върху подходи, които не изискват поставяне на допълнителни устройства върху потребителите. Въпреки че за момента съществуват алгоритми, използващи монокулярни камери, разпознаващи хора и предоставящи информация за скелета им, тенденцията показва, че комбинираният подход – обикновена камера и дълбочинен сензор, дава много по-точни резултати, дори конкурентни на методите, използващи маркери.

След щателно проучване на достъпните технологии установихме, че в Kinect алгоримът за позициониране на ставите в триизмерното пространство е най-ефективно имплементиран. Поради тази причина и качеството на разработката, взехме решението да използваме този сензор.

В раздел „Допълнителни имплементации и тестове“ са приложени тестове с други алгоритми, освен основния за тази разработка.

# Предишна работа

Множество решения на проблема, използващи различни източници на данни, са предложени през изминалите години. Информация за позата е най-често извличана от системи с маркери [1], монокулярни камери [2] [3] [4] [5], силуетни образи [6] и алгоритми, използващи дълбочинна картина [7]. След цялостно проучване, не успяхме да намерим публикувана разработка, използваща Kinect сензор.

На разположение са също и алгоритми, способни да заменят Kinect сензора. Такива са например базираните на TOF камери [8], монокулярни [9] [10] [11] [12], дълбочинни [13] [14] и други [15], които бяха отхвърлени заради качестовото на Kinect сензора.

# Методи

Тази секция цели да изясни подробно методите използвани за реализация на алгоритъма. За леснота, методите са подредени под ред на изпълнение.

## Ъгли и ротация на ставите

След анализ на данните, които бихме могли да извлечем от използвания сензор, взехме решение, че най-добрият подход към проблема би бил да използваме ротацията на ставите в пространството, тъй като тя еднозначно дава информация за движението. [16]

Друг плюс на използването на ротацията на ставите за първостепенна информация за разпознаване на дейностите е, че не се обвързваме с точни координати, а от там – и с точни дължини на костите на скелета.

Разработени са два подхода на представяне на ъглите:

### Представяне на ъглите чрез двуизмерни проекции координатите

Макар и не реална, този метод дава еднозначна информация за ъгъла на ставата.

Същността му се състои в това да групираме двуизмерните прокеции на триизмерните вектори по двойки и да намерим 3 числа, които представят ъгъла на ставата. Нека представим двуизмерните вектори (x-xi, y-yi), (x-xi, z-zi), (y-yi, z-zi) и векторите (xi – xii, yi – yii), (xi – xii, zi-zii), (yi –yii, z-zii). Така, ако намерим двуизмерните ъгли между и , и и и , то ще получим ъглите α, β и γ, чрез които после по-късно бихме могли да сравним дадения ъгъл с друг, но не бихме могли да реконструираме точните променливите, представящи векторите на ставите.

Дясно рамо (x, y, z)

Десен лакът (xi, yi, zi)

Дясна китка (xii, yii, ziii)

,,

,,

Проблем, възникващ при използването на този подход, е създадената зависимост на ориентацията на потребителя спрямо сензора – методът дава различни данни, когато потребителят е лицево ориентиран спрямо сензора, и когато е ориентиран например на 45°. Това би могло да се избегне, като пресмятаме разликата между едната и другата ориентация, което дефинираме като „големина на грешка“ – колкото по-голяма е „големината на грешката“, толкова повече се различават двете ориентации на потребителя.

Можем да пресметнем големината на грешката, като намерим разликата в ориентацията вектора с начало координатите на HipCenter и край координатите на произволно HipLeft или HipRight. Така намираме еднозначно разликата в ротацията на потребителя спрямо сензора.

Въпреки интуитивността на този подход, намирането на точна зависимост на големината на грешката и пресметнатите двуизмерни ъгли, е трудоемък процес, който често дава отклонение, което е критично при използвания алгоритъм, който изисква точна и еднозначна информация за ставите.

### Представяне на ротацията на ставите чрез геометрични кватерниони

Чрез прилагането на тои метод бихме могли да представим ротацията и позицията на ставата чрез геометричен кватернион, който ни служи, както да сравняваме приликата между ротациите, така и да реконструираме позицията и ротацията на ставата.

Под предвид е взето, че системата на кватерниона дава както положителен, така и отрицателен резултат.

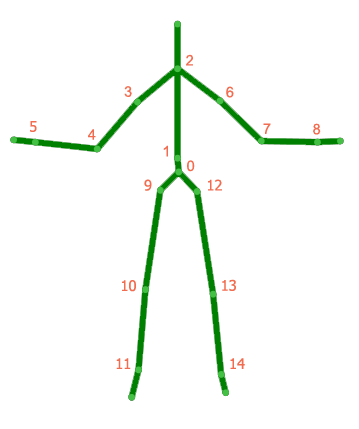
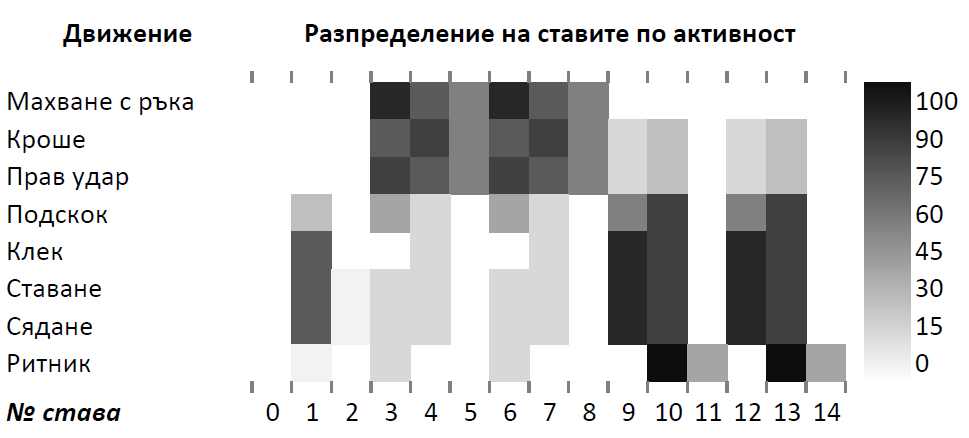
### Заключение

Имплементация на двата представени метода показва, че и двата дават добри резултати в контекста на алгоритъма. За удобство в сегашната си версия алгоритъмът използва втория метод, поради възможността от неточност при използването на първия метод.

## Селекция на най-информативни стави

Минали проучвания показват, че за разпознаване на движения на база ротацията на ставите, не е нужно да се разглеждат всички стави на скелета [16] [17]. Достатъчно е да следим единствено ставите, които имат най-голяма промяна в ротацията си по време на движението. Прост пример би било махането с ръка. Няма смисъл да взимаме предвид ъгъла на левия глезен, за да разпознаем това движение, тъй като ставите, които определят движението са рамото и лакътят. Подкрепящи това твърдение доказателства могат да бъдат намерени на [17] и на фигура 1.

Фигура 1:



Най-активните стави се маркират при предварително записаните движения. На записите се прави анализ, който определя степента на активност по формулата.

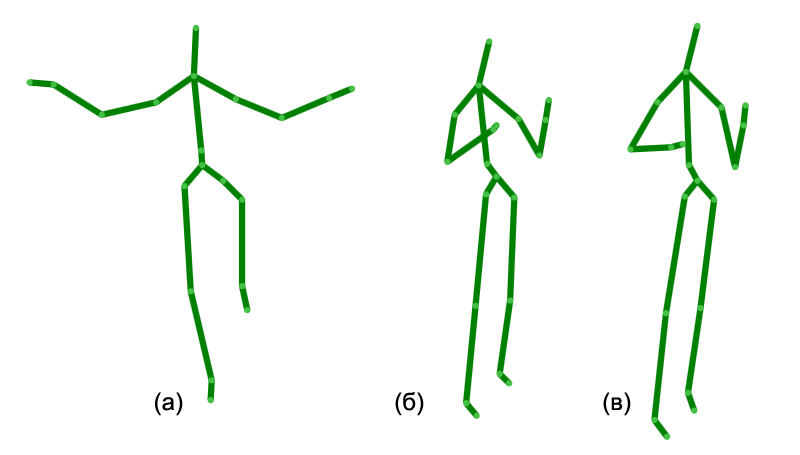
В последствие се селектират единствено „топ-ставите“ или тези, които от които се определя движението. На фигура 2 можете да видите графиката на най-активните стави и точната селекция на най-активните такива.

Фигура 2:

## Сравняване на движения

За да решим глобалния проблем за разпознаване на активности в реално време, първо трябва да решим подпроблема за сравняване на два скелета. Прост пример виждаме във фигура 3а, 3б и 3в. Лесно можем да определим, че фигура 3б прилича много повече на фигура 3в, отколкото на 3а.

Фигура 3:



За да сравним скелетите, които ще означим с A и B, трябва да сравним ротацията на ставите им. В минали версии на разработката, в които се използваха двумерните проекции на координатите на ставите, представяхме трите проектции като координати на точка в триизмерното пространство и изцислявахме евклидовото пространство между точката от първата и втората поза. Това ни предоставяше еднозначен начин да представим приликата между двете стави – колкото по-малко е разстоянието, толкова повече си приличат двете стави по ротация.

В настоящата имплементация на разработката използваме метода за представяне на ротацията на ставите чрез кватерниони (R, S), като използваме следната формула:

Колкото по-голям е полученият косинус, толкова повече сe приближават двата кватерниона.

Друг метод за сравнение на ротацията на ставите чрез кватерниони (R, S) е да използваме следната формула:



## Dynamic time warping (DWT)

Поради липсата на български превод на понятитето, в настоящата разработка ще го наричаме „Динамично изкривяване на времето“ (ДИВ). Този метод служи за измерване на приликата между две редици, които могат да варират спрямо време и скорост. [18] [19] [20]

Чрез прилагането на ДИВ бихме могли да намерим приликата между две активности, дори и ако те са извършени с различни скорости. Например, ако в единия случай потребителят извършва активността „удар с ръка“ с една скорост, а в другия случай я извършва по-бързо, то без да се затрудним бихме могли да открием, че в същността си двете активности представляват едно и също.

Ако не вземем предвид нуждата да разпознаваме активностите в реално време, то бихме могли да дадем следния пример. Записът на активността А има продължителност 50 фрейма, а записът на активността Б има продължителност 100 фрейма. Очевадно скоростта извърешната активност Б е два пъти по-бърза, отколкото скоростта на активността А. Ако приложим линейна проверка (кадър по кадър), резултатът ще бъде грешен – приликата между двата записа би била оценена грешно, което би повлияло на вземането на крайното решение.

В следващия раздел ще разгледаме случая, в който динамичното изкривяване на времето трябва да се приложи в реално време.

Псевдо код на ДИВ в контекста на сравняване на два записа на активности:

|  |
| --- |
| int CompareActivity(Skelton[] s[1..n], Skeleton t[1..m]) {  declare int DynamicTimeWarping[0..n, 0..m]  declare int similarity  for i := 1 to m{  DynamicTimeWarping[0, i] := infinity  }  for i := 1 to n{  DynamicTimeWarping[i, 0] := infinity  }  DynamicTimeWarping[0, 0] := 0  for i := 1 to n{  for j := 1 to m{  similarity:= d(s[i], t[j])  DynamicTimeWarping[i, j] := similarity + minimum(DynamicTimeWarping[i-1, j ],  DynamicTimeWarping[i , j-1],  DynamicTimeWarping[i-1, j-1])  }  }  return DynamicTimeWarping[n, m]  } |

Поради скоростта *О(n\*m)* на този алгоритъм е възможно в бъдещото развитие на „Разпознаване на човешки активности на база 3D реконструкция на скелета“, да бъде имплементиран FastDTW [21], рекурсивен вариант на динамичното изкривяване на времето, който работи със значително по-голяма скорост.

## Разпознаване на активности в реално време

Подходът, нужен за разпознаване на активности в реално време, силно се различава от този, използван за разпознаване на активности в записи, тъй като ДИВ работи с две предварително зададени редици.

За момента предлагаме две решения на проблема.

### Използване на „прозорец”

Дефинираме „прозорец” като последователна поредица от кадри, чието начало е най-новият кадър, а дължината бележим с n. За удобство ще бележим прозореца с W.

Ако на всеки кадър прилагаме ДИВ с параметри W и всеки един от записите в базата данни, то ще постигнем теоритична сложност: 30 движения \* 30 записа \* средно 100 кадъра \* n \* 60 (операции, нужни за сравняване на 2 скелета) = 540000\*n. От тук разбираме, че най-голямата възможна стойност на n би била 100 кадъра, за да бъде разпознаването в реално време. Ако при имплементацията се използва и процесорът на видео картата, дължината на прозореца би могла да достигне до средно 130 кадъра.

Един от вариантите, който предстои да бъде тестван, е да се дават по-малко записи на възможните движения или пък на база всички записи, да се направи „осреднен” вариант на всички тях.

Използването на „прозорец” би могло ни даде добри резултати, но би изисквало голяма хардуерна мощ.

### Използване на маркери за начало и край на движението

Друг възможен подход за решение на проблема, е поставянето на „маркери”, които да означават началото и края на движението, за да може после кадрите между тези маркери да бъдат подадени на ДИВ алгоритъма, който да оцени до колко извършеното движение прилича на някое от записите.

Поставняето на маркери се извършва на база колко скелетът от сегашния кадър прилича на скелет от някой от началните или крайните кадри на записаните движения.

## Заключение

Разпознаването на активности в реално време е сложен проблем с много решения. В настоящата работа са предложени два подхода, чийто недостатък би бил единствено изискването на повече хардуерна мощ от средностатистическата за домашен компютър.

Начин за оптимизация би бил използването на FastDTW, който предстои да бъде имплементиран и тестван за нуждите на проекта, както и имплементирането на настоящия алгоритъм по подходящ начин, за да използва и процесора на графичната карта.Тестове с други алгоритми са извършени в секцията Допълнителни имплементации и тестове. В процес на имплементация е решение на проблема с използването на скритите модели на Марков.

# Имплементация

Най-новото работно копие на имплементацията може да бъде намерено в хранилището на проекта на адрес <http://goo.gl/0y6t4>

## Използвани технологии

### Kinect SDK

Kinect SDK 1.5.1 е използван за получаване на данни от Kinect сензора. Заедно с Kinect SDK е използвано също и Kinect Studio, кoето позволява възпроизвеждането на записи в апликацията.

### C# & .NET

Имплементацията на алгоритъм от този сорт се нуждае от бърз достъп до данни, добре организирани чрез обектно-ориентиран подход на програмиране. ООП също предоставя модуларността и сигурността на имплементацията. .NET и C# предоставят лесен достъп до данните от Kinect сензора.

# Възможни приложения

Прилагането на алгоритъма в реални условия би спомогнало за подобрението на качеството на услугите в широк спектър от сфери. От видеонаблюдение в магазини, където се следи за „съмнителни“ активности – например прибиране на нещо в джоба, видеонаблюдение на паркинги, където се следи как точно се отваря врата – с ключ или по друг начин и т.н. Друго приложение би било интелигентна класификация на филми на база съдържанието им. Особен акцент трябва да се сложи върху системите за наблюдение на пациенти, които могат да следят за най-различни активности – от сигнал за викане на сестра, до вземане на хапчета, преминаване от стая в стая, степен на активност и т.н.

Друго приложение, към което се стреми голяма част от общността на учените в момента, e изграждането на цялостен NUI интерфейс или начин за влизане във взаимодействие с компютъра по естествени пътища – не чрез мишка и клавиатура, а чрез говор, действия и други. Чрез разработения алгоритъм лесно би могло да се разработи приложение, което бива управлявано единствено чрез човешки действия.

Системите за „умни домове“ все повече навлизат в нашето ежедневие. Чрез използването на предложения алгоритъм лесно може да се разширят възможностите им, като по този начин улеснят още повече ползвателите им.

# Допълнителни имплементации и тестове

## Minimal Variance Matching (MVM)

В изследването на учени от Temple university, Филаделфия е предложен подход за намиране на разлика между две различни по дължина редици. (фигура 4)

Подходър разчита на генерирането на матрица на разликата на всеки два елемента от двете редици, като в по-късното сравнение на редиците се взема предвид и възможността от появата на малки разлики или шум (guassian noise). По-късно, на база на генерираната матрица на разликите, бива построенен асинхронен цикличен граф, в който в последствие се прилага алгоритъм за намиране на най-кратък път, който дава еднозначно разликата между двете редици.

В изследването е предложен и подходящ метод за намиране на нужното скалиране и изместване на едната редица спрямо другата, за оптимални резултати.

Поради многото мнения в обществото на учените, че този алгоритъм би могъл да даде по-добри резултати в контекста на настоящата разработка. Въпреки факта, че MVM допуска пропускането на елементи от едната редица в другата, той бе имплементиран и тестван. (таблица 1)

ДА ДОБАВЯ ТАБЛИЦА 1 – сравнение между Elastic matching & DTW

## Алгоритъмът, който измислих!

# Заключение

Преложеният алгоритъм решава поставения проблем чрез използването на ефективен подход за подбор на данни, при който те придобиват интуитивно физическо значение. За разпознаването на действия в реално време е използван алгоритъм за динамично изкривяване на времето, който предоставя релативно ефективен начин за разпознаване на еднакви движения, извършени с различни скорости. С цел по-високо ниво на качество, за набавяне на данни за триизмерния скелет на потребителя, се използва Kinect сензор. По-голяма точност на разпознаване на движенията се постига чрез филтриране на входните данни за скелета.

За в бъдеще би могло да се оптимизира времето на работа на алгоритъма за динамично изкривяване на времето, както и ресурсите, които използва.

Благодарности

Авторът желае да изкаже благодарност на Румен Данговски, Валерия Станева, Калина Петрова и ръководителя на проекта, Светослав Колев, за ценните съвети и изказани мнения. Също така желае да изкаже благодарност и към Ученическия институт по математика и информатика (УчИМИ) към БАН, който организира „Лятна изследователска школа”, където бе поставено началото на проекта.

# Използвана литература

|  |  |
| --- | --- |
| [1] | R. J. S. S. S. R. B. Allen Y. Yang, "Distributed Recognition of Human Actions Using Wearable Motion Sensor Networks," *Journal of Ambient Intelligence and Smart Environments,* 2009. |
| [2] | N. P. Osama Masoud, "A Method For Human Action Recognition," 2003. |
| [3] | H. I. Yasuo Kuniyoshi, „Qualitative Recognition of Ongoing Human Action Sequences“. |
| [4] | M. M. U. A. K. I. L. C. S. Heng Wang, „Evaluation of local spatio-temporal features“. |
| [5] | K. H. T. T. Shiquan Wang, „Human action recognition with pose similarity“. |
| [6] | Z. Z. Z. L. Wanqing Li, „Expandable Data-Driven Graphical Modeling of Human Actions Based on Salient Postures,“ *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY,* том 18, 2008. |
| [7] | Z. Z. Z. L. Wanqing Li, „Action Recognition Based on A Bag of 3D Points“. |
| [8] | F. M. Ayet Shaiek, "3D keypoint detectors and descriptors for 3D objects recognition with TOF camera," *IS&T/SPIE Electronic Imaging conference on 3D Image Processing (3DIP) and applications,,* 2011. |
| [9] | J. C. Yen-Lin Chen, „3D Reconstruction of Human Motion and Skeleton from Uncalibrated Monocular Video“. |
| [10] | V. L. a. P. F. M. Dimitrijevic, „Human Body Pose Recognition Using Spatio-Temporal Templates“. |
| [11] | B. Laxton, „Monocular Human Pose Estimation“. |
| [12] | P. L. P. F. Vincent Lepetit, „Randomized Trees for Real-Time Keypoint Recognition“. |
| [13] | A. F. M. C. T. S. M. F. R. M. A. K. A. B. Jamie Shotton, „Real-Time Human Pose Recognition in Parts from Single Depth Images“. |
| [14] | A. S. Himanshu Prakash Jain, "Real-time Upper-body Human Pose Estimation using a Depth Camera". |
| [15] | J.-M. O. Jian Yao, „Fast Human Detection from Joint Appearance and Foreground Feature Subset Covariances,“ *CVIU,* том 115, № 10, 2011. |
| [16] | J. G. G. F. L. V. G. Angela Yao, „Does Human Action Recognition Benefit from Pose Estimation?“. |
| [17] | R. C. G. K. R. V. a. R. B. Ferda Ofli, „Sequence of the Most Informative Joints (SMIJ): A New Representation for Human Skeletal Action Recognition“. |
| [18] | M. Müller, Information Retrieval for Music and Motion, Springer, 2007. |
| [19] | P. Senin, "Dynamic Time Warping Algorithm Review," 2008. |
| [20] | R. Niels, „Dynamic Time Warping - An intuitive way of handwriting recognition?,“ 2004. |
| [21] | P. C. Stan Salvador, „FastDTW: Toward Accurate Dynamic Time Warping in Linear Time and Space“. |
| [22] | Z. Z. Z. L. Wanqing Li, „Expandable Data-Driven Graphical Modeling of Human Actions Based on Salient Postures,“ *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY,* том 18, 2008. |

ДА ДОБАВЯ Elastic matching!