УДК 004.9

Ю.В. Крак, А.С. Тернов, Д.С. Ульянич

Анализ мануальных компонентов украинской жестовой речи с использованием системы дополнительных маркеров

Рассмотрено создание информационной технологии распознавания отдельных жестов украинского жестового языка, полученных посредством технологии *Motion Capture*. Методами исследования служат алгоритмы динамической свертки временной последовательности.

Ключевые слова: жестовый язык, распознавание, информационная технология, захват движения, динамическая свертка временной последовательности с использованием весов.

Розглянуто створення інформаційної технології розпізнавання окремих жестів української жестової мови, отриманих за технологією *Motion Capture*. Методами дослідження слугують алгоритми динамічної згортки часової послідовності.

Ключові слова: жестова мова, розпізнавання, інформаційна технологія, захват рухів, динамічна згортка часової послідовності з використанням ваг.

Введение. Для распознавания жестов используются методы компьютерного зрения (*Computer Vision*), позволяющие позиционировать движущиеся объекты в пространстве [1]. Поскольку жестовый язык изучается в основном в специализированных учебных заведениях Украины, то возникает проблема общения между глухими и слышащими людьми [2]. Эту проблему пытаются решить путем создания информационной технологии *понимания* жестов [3]. Для распознавания жестовой речи (ЖР) требуется идентификация жестов через динамику движений. При этом необходимо распознавать положение рук на изображении и учитывать изменение формы кисти руки, мимику лица, расположение рук относительно друг друга и относительно туловища и лица. Алгоритмы обработки изображений должны быть устойчивыми к перекрытию рук и лица, адаптироваться к работе с новым пользователем.

Большинство систем распознавания ЖР используют упрощенные характеристики, а словари систем выбирают таким образом, чтобы минимизировать количество схожих жестов [4–7]. Исследование методов построения систем распознавания ЖР – важная и актуальная проблема.

Постановка задачи

Цель статьи – создание универсальной информационной технологии распознавания одиночных украинский жестов речи, снятых по технологии захвата движений *Motion Capture*.

Алгоритмы анализа

Основная идея в классификации данных заключается в одновременном уменьшении внутри классовой и увеличении межклассовой вариации данных (*Within-class and Between-class Variation*). В статье роль вариации (отклонения) данных принадлежит расстоянию (*Dynamic Time Warping – DTW*). В дальнейшем термины образец, экземпляр, жест и временная последовательность считаются синонимами, и все они касаются некоторой последовательности данных.

Алгоритм динамической трансформации временной шкалы позволяет найти оптимальное соответствие между временными последовательностями. Для вычисления отклонения достаточно измерения расстояния между компонентами двух последовательностей (евклидово расстояние). Однако не всегда две последовательности, имеющие одинаковую общую форму, выровненные по временной шкале. Чтобы определить сходство между такими последовательностями, нужно *деформировать* шкалу времени одной (или обеих) последовательности, для достижения лучшего выравнивания. Использование евклидова расстояния имеет существенный недостаток: если два временных ряда одинаковые, но один из них немного смещен

во времени (вдоль временной оси), то евклидова метрика может посчитать, что ряды отличаются друг от друга. *DTW*-алгоритм был введен для того, чтобы преодолеть этот недостаток и предоставить наглядное измерение расстояния между рядами, не обращая внимание как на глобальные, так и на локальные сдвиги по временной шкале.

Рассмотрим две временные последовательности Q длиной n и C длиной m: $Q = q_1, q_2, \ldots, q_n$, $C = c_1, c_2, \ldots, c_m$. Вычисление *DTW*-расстояния проходит по следующим этапам:

• Вычисляется матрица d размерности $m \times n$ (матрица расстояний), в которой элемент d_{ij}, $i = \overline{1,n}$, $j = \overline{1,m}$ есть расстояние $d(q_i.c_j)$ между двумя точками q_i и c_j. Обычно используется евклидово расстояние: $d(q_i.c_j) = (q_i - c_j)^2$ или $d(q_i.c_j) = |q_i - c_j|$.

• Строится матрица трансформаций (деформаций) D, каждый элемент которой вычисляется следующим образом: $D_{i,j} = d_{i,j} + \min(D_{i-1,j}, D_{i-1,j-1}, D_{i,j-1})$.

• Определяется оптимальный путь трансформации (деформации) $W = w_1, w_2, \ldots, w_k$, где k-й $w_k = (i,j)_k$, и *DTW*-расстояние $d(w_k) = d(q_i, c_j)$. Таким образом, $\max(m,n) \le K < m+n$, где K – длина пути.

Следует отметить, что путь трансформации W содержит все точки обоих временных рядов, передвигается не более чем на один шаг за один раз и не возвращается назад к уже пройденной точке.

DTW-расстояние или стоимость пути между двумя последовательностями рассчитывается на основе оптимального пути трансформации с помощью формулы:

$$DTW(Q,C) = \min_{i,j}\left\{\sum_{i=1}^{n}\sum_{j=1}^{m} d(q_i,c_j) \Big/ K\right\}. \quad (1)$$

Длина оптимального пути K используется для учета путей свертки разной длины.

Преимущества *DTW*-алгоритма:

– результат сравнения не зависит от скорости воспроизведения и длины представления двух сравниваемых временных последовательностей;

– простой в реализации;

– не зависит от количества классов данных.

Недостаток *DTW*-алгоритма – его применение на реальных данных, так как требуется сглаживание или фильтрация.

Алгоритм динамической свертки временной последовательности с использованием весов (*Weighted Dynamic Time Warping – WDTW*) есть модификацией классического варианта *DTW* и наследует все его преимущества и недостатки. Модификация заключается в следующем.

Пусть необходимо классифицировать часовую последовательность Q объекта P, состоящего из двух подвижных частей a и b, каждая из которых двигалась в некотором пространстве в течение фиксированного времени tQ, с другой, временной последовательностью C длиной tC того же объекта P. Для этого находится *DTW*-расстояние между Q и C для каждой подвижной части и суммируется результат:

$$DTW(Q,C) = DTW(Q_a, C_a) + DTW(Q_b, C_b). \quad (2)$$

Использование весов, а именно весовой динамической свертки временной последовательности (*WDTW*), позволяет внести дополнительную информацию об объекте P, который перемещается в пространстве:

$$WDTW(Q,C) = w_a \cdot DTW(Q_a, C_a) + w_b \cdot DTW(Q_b, C_b), \quad (3)$$

где $w_a + w_b = 1$, $|w_a| < 1, |w_b| < 1$.

Использование весов, в общем случае, нарушает симметричность алгоритма с аргументом, т.е. $WDTW(Q,C) \ne WDTW(C,Q)$, поскольку веса маркеров для последовательностей Q и C могут быть разными.

Быстрый алгоритм динамической свертки временной последовательности (*Fast Dynamic Time Warping*, далее – *FastDTW*) использует идеи ограничения и абстракции данных. Использование комбинации обоих вышеупомянутых категорий устраняет недостатки при их отдельном использовании и принимает линейную сложность как во временном, так и в пространствен-

ном масштабе. *FastDTW*-алгоритм использует многоуровневый подход с тремя ключевыми операциями [8]:

– уменьшение детализации – уменьшение длины входной последовательности таким образом, чтобы выделенное представление данных максимально отражало входные данные;

– проекция – нахождение пути с минимальной стоимостью на уменьшенном представлении данных и использование найденного пути для построения улучшенного результата на предварительных данных большей длины;

– уточнение – уточнение пути деформации, проектируемого с данных с уменьшенной детализацией, путем локальных модификаций найденного пути на начальных данных с большей (полной) длиной. Реализация и псевдокод *FastDTW* приведены в [8].

Модель данных

Для распознавания ЖР необходимо четко установить положение руки относительно тела человека и, что более характерно именно для украинского жестового языка, положение пальцев рук [4]. Для этого создан проект «УкрЖест», который содержит 139 распространенных жестов, полученных по технологии *Coordinate* 3*D Motion Capture* [9].

Продолжительность одного жеста не превышает пяти секунд, другие параметры данных были следующими: тип данных – *.c3d*, размерность – 3*D*, измерение – миллиметр, количество маркеров – 83, количество несущих маркеров – 50, кадровая частота данных (*FPS*) – 120, количество уникальных классов данных – 139 (рис. 1). Так на теле человека (актера) были размещены 83 оптических датчика с большим их количеством в областях кисти руки. На каждой руке было установлено 25 датчиков. Такое чрезмерное количество датчиков на одну руку обусловлено той особенностью украинского ЖР, что для корректной идентификации жеста необходимо четко установить положение пальцев рук.

Для записи видеофрагментов жестов использовались 16 камер *Vicon Bonita*, расположенных вокруг актера с рабочей зоной восемь на восемь метров. Каждая камера имеет разрешение матрицы 1 Мп, что позволяет снимать с частотой 120 кадров в секунду с точностью позиционирования маркеров до 0,5 мм.

Рис 1. Демонстрация записи жеста «День»

Модуль подготовки данных включает:

- вычитание центральной точки между плечами от всех координат маркеров для учета случаев, когда актер менял свое местонахождение между демонстрацией жестов;
- деление координат маркеров на длину между плечами для учета актеров с различными размерами тела.

Это было сделано с целью генерализации данных (абстракции от физических факторов съемки).

Точность, достигнутая при получении данных с использованием профессионального оборудования, позволила избежать этапа очистки данных от лишних шумов.

Выбор *WDTW*-алгоритма за основу обусловлен особенностью базы данных проекта УкрЖест, поскольку обучающая выборка состоит только из одного экземпляра на отдельный класс данных. Этот факт существенно ограничивает использование мощных систем распознавания (как, например, искусственные нейронные сети), которым необходимо большое количество учебных данных и у которых в результате низкий уровень генерализации данных.

Схема информационной технологии

Блок-схема распознавания (идентификации) отдельного жеста представлена на рис. 2. На

первом этапе вызывается обработчик данных, считывающий положение маркеров в пространстве по времени, наименование присутствующих маркеров, кадровую частоту и другие данные, необходимые на следующем этапе.

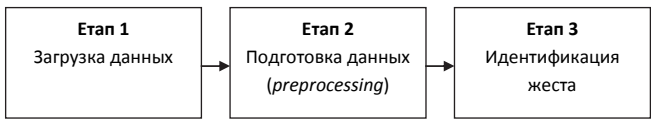

Рис 2. Общая схема распознавания одиночного жеста

На *втором* этапе происходит подготовка данных (*data preprocessing*). Он необходим для того, чтобы алгоритм идентификации не был зависим от особенностей съемки жеста и персонажа, воспроизводящего его при съемке.

На *третьем* этапе происходит идентификация предварительно обработанных (подготовленных) данных выбранным алгоритмом классификации. На выходе из третьего этапа имеем наименование класса данных, к которому, наиболее вероятно, по решению выбранного алгоритма классификации относится неизвестный нам жест.

Для алгоритма динамической свертки временной последовательности маркеров необходимо задать функцию *расстояния* между двумя любыми моментами i, j – для двух сравниваемых временных последовательностей: q_i и c_j. В данном случае считается евклидово расстояние $d_m(q_i, c_j)$ для каждого присутствующего маркера в обоих временных последовательностях: $d_m(q_i, c_j) = \left\| q_i^m - c_j^m \right\|$. Тогда *DTW*-расстояние между двумя любыми ячейками i и j считается как суммарное отклонение всех присутствующих маркеров: $dtw(q_i, c_j) = \sum_{m=1}^{M} d_m(q_i, c_j)$.

Отметим, что в таком представлении отклонение любого маркера есть независимым и равноценным, т.е. теряется информация о несущих маркерах, а значит о тех уникальных маркерах, совокупность которых точно определяют воспроизводимый жест. Например, в украинском ЖР определяющую роль несут положение пальцев каждой руки. Поэтому, для более эффективной работы алгоритма динамической свертки временной последовательности, предложено использовать весы, определяющие активность каждого маркера [10]. Такая существенная модификация алгоритма позволяет ориентироваться на движение маркеров с большим весом.

Измерение активности маркеров

Расчет веса для m-го маркера уникального жеста g проводится по следующей формуле:

$$w_m^g(\beta) = \frac{1 - e^{-\beta D_m^g}}{\sum_k \left(1 - e^{-\beta D_k^g}\right)}, \qquad (4)$$

где D_m^g – суммарное отклонение (активность) маркера m для жестового класса g, усредненное по обучающей выборке, β – скрытый параметр.

Отметим, что суммарное отклонение маркера m считается так:

$$D_m = \sum_{i=2}^{N} \left\| \vec{X}_i^m - \vec{X}_{i-1}^m \right\|, \qquad (5)$$

где $\vec{X}_i^m = (x_i^m, y_i^m, z_i^m)$ – позиция m-го маркера на i-м кадре (фрейме).

Скрытый параметр β определяется путем максимизации величины различия (*discriminant ratio*) $R = D_b / D_w$, где D_b и D_w – соответственно межклассовая и внутри классовая вариация (*Between-class and Within-class Variation*). Межклассовая вариация D_b вычисляется как усредненная *WDTW*-расстояние между двумя экземплярами q_i и c_j из разных классов данных,

$$D_b = \left\langle \sum_{Q_k} \sum_{Q_p \neq Q_k} \sum_{q_i \in Q_k} \sum_{c_j \in Q_p} wdtw(q_i, c_j) \right\rangle, \qquad (6)$$

а D_w вычисляется как усредненная *WDTW*-расстояние между двумя разными экземплярами из одного класса данных:

$$D_w = \left\langle \sum_{Q_k} \sum_{q_i \in Q_k} \sum_{c_j \in Q_p : c_j \neq q_i} wdtw(q_i, c_j) \right\rangle, \qquad (7)$$

где суммирование проводится по всем классам данных Q_k.

Пусть q – известная временная последовательность (жест) из класса данных Q, c – неизвестная временная последовательность, которая сравнивается с известной временной последовательностью. Тогда *WDTW*-алгоритм отличается от *DTW*-алгоритма только функцией

расстояния между моментами (фреймами) i_1 и i_2 временных последовательностей q и c:

$$wdtw(q_{i_1}, q_{i_2}) = \sum_{m=1}^{M} d_m\left(q_{i_1}, q_{i_2}\right) \cdot w_m^Q. \quad (8)$$

Несущими маркерами выбраны маркеры обеих рук, диаграмму активности которых приведены на рис. 3, где более светлые маркеры соответствуют области наибольшей активности.

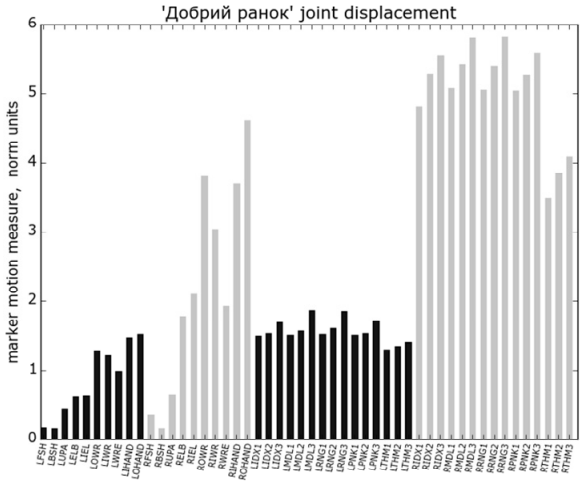

Рис 3. Активность несущих маркеров при воспроизведении жеста «Доброе утро»

Этап обучения *WDTW*-алгоритма проводится по следующему сценарию:

1. Выбрать произвольное β.
2. Вычислить матрицу весов по формуле (2).
3. Вычислить внутриклассовую вариацию D_w.
4. Вычислить межклассовую вариацию D_b.
5. Найти величину различия $R(\beta) = D_b/D_w$.
6. Повторить шаги 1–4 для каждого $\beta \in \{\beta_1, ..., \beta_n\}$.
7. Определить оптимальное значение $\beta_{opt} = \arg\max_\beta (R(\beta))$.

Оптимальное значение β_{opt} находится из набора $\beta = e^a, \forall a \in \{-6, -4, -2, -1, 0, 1, 2, 3, 4\}$ (рис. 4).

Анализ полученных результатов позволил сделать следующие выводы:

1. При очень малых и очень больших значениях β величина различия $R(\beta)$выходит на уровень насыщения.

2. При больших β исчезает информация об активности маркеров:

$$w_m^g \underset{\beta \to \infty}{\to} \frac{1}{M}, \quad (9)$$

где M – общее количество (несущих) маркеров. В таком случае значение весов маркеров – константа, и нет смысла использовать *WDTW*-алгоритм, т.е. при $\beta \to \infty$ *WDTW*-алгоритм сводится к обычному *DTW*-алгоритму.

3. Обучающая выборка состоит только из одного экземпляра на один класс данных, т.е. нет возможности вычислять межклассовую вариацию данных D_w.

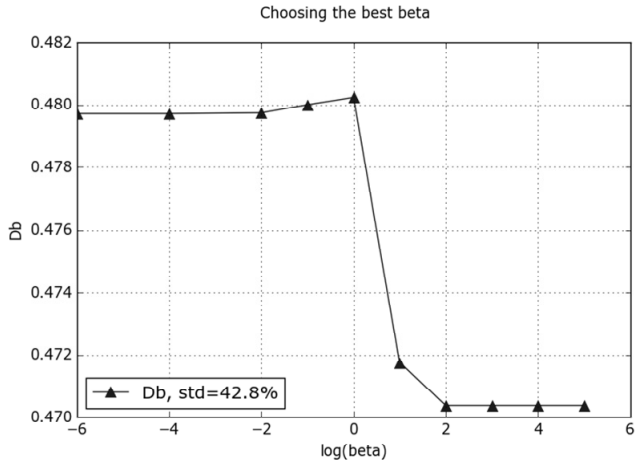

Рис. 4. Подбор оптимального параметра $\beta_{opt} = \arg\max_\beta (R(\beta))$

Таким образом, выбор оптимального параметра $\beta_{opt} \to 0$ завершает этап обучения.

Этап тестирования проводился по наихудшему сценарию. Для каждого неизвестного образца c, принадлежащему классу данных Q, из тестовой выборки вычисляется *WDTW*-расстояние со всеми экземплярами обучающей выборки, принадлежащих или не относящихся к классу Q неизвестного образца. Обозначим их соответственно q_i и h_j.

Далее сравнивается максимальное *WDTW*-расстояние между c и каждым q_i с минимальным *WDTW*-расстоянием между c и каждым h_j. Если выполняется условие

$$\max_i wdtw(c, q_i) < \min_j wdtw(c, h_j), \quad (10)$$

то образец c считается идентифицированным верно.

Параллельно с этим вычисляется ошибка за лучшим (или классическим) сценарием. В этом случае условие (10) имеет вид:

$$\min_i wdtw(c, q_i) < \min_j wdtw(c, h_j). \quad (11)$$

Если ошибка по лучшему сценарию не совпадает с ошибкой по наихудшему, это означает,

что существует перекрытие областей решений. Тогда критерием успешности алгоритма выступает относительное количество верно идентифицированных образцов из тестовой выборки от общего количества образцов в тестовой выборке, т.е. уровень распознавания тестовых данных.

Обсуждение результатов

Результаты распознавания тестовых данных *WDTW*-алгоритмом в сравнении с классическим *DTW*-алгоритмом продемонстрировали, что оба алгоритма распознали все тестовые данные проекта УкрЖест, а потому для него модификация весов *сработает* при любом значении β (2). Это связано с особенностями его базы данных:

- все записанные жесты воспроизводились опытным сурдопереводчиком;
- частота записи данных в 120 кадров – чрезмерна, поскольку для *WDTW* вполне достаточно работать с временными последовательностями с частотой восемь кадров в секунду (рис. 5);
- из 83 маркеров, размещенных на сурдопереводчике, используется 50 маркеров (по 25 на каждую руку).

Открытым остается вопрос наименьшего количества маркеров (и их расположение), не ухудшающие результат распознавания.

Для решения проблемы уменьшения данных временной последовательности, при условии не ухудшения результатов распознавания *WTDW*-алгоритмом, разработана функция уменьшения начальной кадровой частоты данных. В результате построена зависимость ошибки по наихудшему сценарию алгоритма *WTDW* на тестовой выборке *E_test* от кадровой частоты данных *FPS* (рис. 5).

Представленная зависимость *E_test* (*FPS*) остановлена на частоте $FPS = 10$, поскольку при дальнейшем увеличении параметра *FPS* величина ошибки *E_test* не меняется и равна нулю. Отметим, что уменьшение *FPS* в 15 раз не ухудшает результатов распознавания *WDTW*-алгоритмом, но при дальнейшем уменьшении кадровой частоты ($FPS < 8$) начинают возникать ошибки распознавания жестов. Таким образом, уменьшение кадровой частоты до величины $FPS = 8$ не влияет на результат распознавания *WDTW*-алгоритмом.

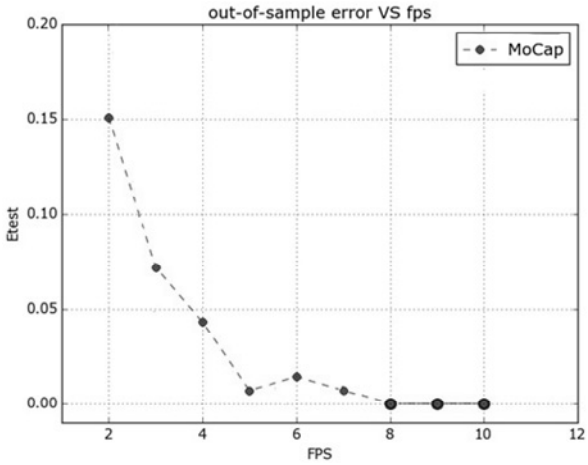

Рис. 5. Зависимость ошибки алгоритма от кадровой частоты данных

Отметим, что предложенный *WDTW*-алгоритм использует ускорение *FastDTW*, что позволило привести квадратичную часовую и пространственную сложность классического *DTW*-алгоритма к линейной. При этом, как показано в [8], *FastDTW* ускорения уменьшает вероятность нахождения оптимального пути *DTW* свертки, но для данной задачи результаты распознавания *DTW*-алгоритмом в точности совпадают с результатами распознавания алгоритмом *FastDTW*.

Заключение. В статье предложено построение информационной технологии распознавания отдельных жестов украинского жестового языка с применением алгоритмов динамической свертки временной последовательности. Информационная модель жестов получена с помощью технологии *Motion Capture*. На ограниченных словарях жестов данный подход зарекомендовал себя с положительной стороны (100 процентов образцов из тестовой выборки).

К недостаткам рассмотренных алгоритмов можно отнести достаточно большое время распознавания отдельного образца, поэтому дальнейшие исследования будут направлены на оптимизацию по времени предложенных алгоритмов распознавания жестов.

1. *Holte M., Moeslund T.* Gesture recognition using a range camera // Tech. Rep. CVMT-07-01. – 2007. – С. 1–5.

2. *Кульбіда С.В.* Українська жестова мова як природна знакова система: Зб. наук. праць «Жестова мова й сучасність». – К.: Педагогічна думка, 2009. – С. 218–239.
3. *Крак Ю.В.*, *Тернов А.С.*, *Лісняк М.П.* Розробка архітектури та основних інструментів комп'ютерної анімації для побудови системи синтезу жестової мови // Штучний інтелект. – 2013. – № 3(61). – С. 147–153.
4. *Комп'ютерне* розпізнавання жестів: програмно алгоритмічний підхід: Монографія / О.В. Годич, М.В. Давидов, Ю.В. Нікольський та ін. – Львів: Компанія «Манускрипт», 2011. – 316 с.
5. *Hand* in hand: automatic sign language to english translation / D. Stein, P. Dreuw, H. Ney et al. // Proc. of the 11th Int. Conf. on Theoretical and Methodological Issues in Machine Translation (TMI 2007). – 2007. – P. 214–220.
6. *Zahedi M.*, *Keysers D.*, *Ney H.* Appearance-based recognition of words in american sign language // Iberian Conf. on Patt. Recog. and Image Analysis'05. – 2005. – P. 511–518.
7. *Speech* recognition techniques for a sign language recognition system / P. Dreuw, D. Rybach, T. Deselaers et al. // ISC A best student paper award Interspeech. – Aug. 2007. – P. 2513–2516.
8. *Salvador S.*, *Chan P.* Toward accurate dynamic time warping in linear time and space // Intelligent Data Analysis. – 2007. – **11**, N 5. – P. 561–580.
9. *The 3D* Biomechanics Data Standard. – https://www.c3d.org/
10. *Gesture* Recognition Using Skeleton Data with Weighted Dynamic Time Warping / Sait Celebi, Ali S. Aydin, Talha T. Temiz et al. // Proc. of the Int. Conf. on Comp. Vision Theory and Appl. (VISAPP 2013). – 2013. – **1**. – P. 620–625.

Поступила 09.04.2015
E-mail: *krak@unicyb.kiev.ua*, *anton.ternov@gmail.com*, *dizcza@gmail.com*

UDC 004.93

Iu.V. Krak, A.S.Ternov, D.S. Ulianych

Analysis of the Movement Components of Sign Ukrainian Broadcasting Using the Additional Markers System

Keywords: sign language, recognition, information technology, motion-caption, weighted dynamic time warping.

Introduction. Gesture recognition is an actual problem concerning the interaction between a user and the computers. Although some prototypes of foreign sign language recognition systems have been developed and already used in computer vision, Ukrainian sign language recognition still remains the problem.

Purpose. The aim is to develop a universal recognition technology for single Ukrainian signs captured with Motion Capture technology. The objects of the study are data bases of single signs, presented as time sequences of coordinates of motion components. The research methods cover the dynamic time warping algorithms.

Results. Data model for gesture recognition in small Ukrainian sign language dictionaries is investigated. A DTW (Dynamic Time Warping) algorithm is used. The modified WDTW (Weighted Dynamic Time Warping) using weights that the more an individual marker for moving it in space during the whole demonstration gesture are proposed. The criterion of success algorithm performed relative number of correctly identified samples from a test one to their total number. Both algorithms are demonstrated a full recognition accuracy on the basis of 139 unique gestures. For WDTW algorithm was used normalization, and therefore the results have a high level of generalization.

Conclusions. The WDTW algorithm to create a universal information technology of identification the Ukrainian sign language gestures is used. The gestures in the form of time series calculated using the technology Motion Capture are presented. This frame rate of 10 fps is sufficient for correct identification sign in a limited vocabulary.

●