

ОНТОЛОГИЧЕСКАЯ МОДЕЛЬ РАСПОЗНАВАНИЯ ЖЕСТОВ РУК ОПЕРАТОРА

М. В. Бобырь¹ [0000-0002-5400-6817], А. А. Асеев² [0009-0007-8271-7660]

^{1, 2}Юго-Западный государственный университет, г. Курск, Россия

¹maxbobyry@gmail.com, ²asseeff.artem@gmail.com

Аннотация

Представлена онтологическая модель распознавания жестов рук оператора, основанная на методе скелетизации кистей рук библиотеки MediaPipe Hands и авторском способе когнитивной трансформации состояний пальцев по ключевым точкам скелетизации в бинарный код с последующей его конвертацией в цифровой сигнал управления. Модель системы включает этапы предобработки видеокadra, скелетизации кисти, анализа координат ключевых точек, бинарного кодирования состояний пальцев и формирования итогового десятичного управляющего сигнала. Улучшение технических и методологических характеристик системы достигается за счет явного задания причинно-следственных связей между объектами и процессами в онтологии, а также применения четких правил бинарного кодирования. Разработанный интерфейс распознавания жестов обеспечивает точность 98% в нормальных условиях и не ниже 94% в условиях шума и низкой освещенности (50 лм и менее люкс) при среднем времени отклика 0.5 с, превосходя базовые методы на основе известных библиотек OpenCV и TensorFlow.

Ключевые слова: распознавание жестов, онтология, MediaPipe, алгоритм скелетизации, бинарный код, графический интерфейс, когнитивность.

ВВЕДЕНИЕ

Системы распознавания жестов обеспечивают интуитивное взаимодействие человека с машиной и находят применение в робототехнике, например в задачах дистанционного управления промышленными установками [1, 2]. Подобные системы позволяют передавать управляющие сигналы без использования традиционных устройств ввода, что особенно востребовано в условиях,

требующих бесконтактного взаимодействия. Подобные системы включают видеокамеру и персональный компьютер (ПК). Камера в режиме реального времени фиксирует жесты оператора и передает видеопоток на ПК, где выполняется обработка и формируется цифровой управляющий сигнал, например сигнал управления пропорционально-интегрально-дифференциального (ПИД)-регулятора [3]. Например, оператор жестом задает температуру в помещении, распознанный посредством видеокамеры сигнал поступает в ПК, который формирует сигнал управления ПИД-регулятора и обеспечивает поддержание заданной температуры в помещении. Однако практическая реализация такой системы сопряжена с рядом технических и методологических ограничений.

Так, существующие алгоритмы скелетизации кистей рук, реализованные в библиотеках MediaPipe Hands и TensorFlow [4, 5], используют и определяют только координаты ключевых точек на кадре. В связи с этим возникает задача когнитивной интерпретации полученных координат, т. е. задача сопоставления взаимного расположения точек реальным жестам (определение того, поднят палец или согнут) [6, 7]. Отмечено также, что традиционные решения на основе OpenCV менее устойчивы к шуму видеопотока [8], а системы на базе TensorFlow при высокой точности требуют значительных вычислительных ресурсов [9–11]. При этом проектирование системы без формализованного описания взаимосвязей компонентов порождает дополнительную сложность: инженеру необходимо точно знать, какие параметры и в каком месте управляющего программного кода требуется изменить при замене камеры или расширении набора жестов.

Перечисленные ограничения обусловили необходимость решения задачи разработки онтологической модели распознавания жестов рук оператора (ОМРЖРО). Онтология формализует состав системы (оператор, камера, ПК), типы сигналов, вычислительные и физические процессы, а также причинно-следственные связи между ними при заданной точности. На основе онтологической модели реализована двухуровневая система обработки информации. На первом уровне библиотека MediaPipe Hands [12–14] выполняет скелетизацию кистей рук. На втором уровне разработанный авторский алгоритм когнитивной интерпретации сопоставляет расположение ключевых точек скелетизации жестам,

преобразуя состояния пальцев в бинарный код, и формирует итоговое десятичное число для передачи на исполнительные устройства мехатронных комплексов. Например, жест «ОК» используется для интуитивно понятного подтверждения (фиксации) сформированного значения перед передачей его в систему управления.

ОНТОЛОГИЧЕСКАЯ МОДЕЛЬ РАСПОЗНАВАНИЯ ЖЕСТОВ РУК ОПЕРАТОРА

Онтологическая модель (ОМ) распознавания жестов рук оператора организует структуру секелетизации жестов рук с помощью MediaPipe и интерпретацию в цифровые значения ключевых точек скелетизации с помощью авторского алгоритма. Организация достигается описанием взаимодействия между ее компонентами. ОМРЖРО задается в виде кортежа

$$O = \langle Oo_{i=1}^o, Op_{i=1}^p, Oi_{i=1}^i \rangle, \quad (1)$$

где Oo – ОМ классов объектов ($o = 1 \div 6$, где o – классы объектов); Op – ОМ процесса ($p = 1 \div 13$, где p – подклассы процесса); Oi – ОМ изображения ($i = 1 \div 11$, где i – атрибуты изображения). Модель определяет состав объектов системы, сигналов и процессы их преобразования (табл. 1), а также связи между ними (рис. 1).

Онтологическая модель (1) выделяет как физические объекты (камера, оператор, персональный компьютер), так и программные компоненты (MediaPipe, анализатор пальцев, графический интерфейс), реализующие вычислительные процессы обработки сигналов [15].

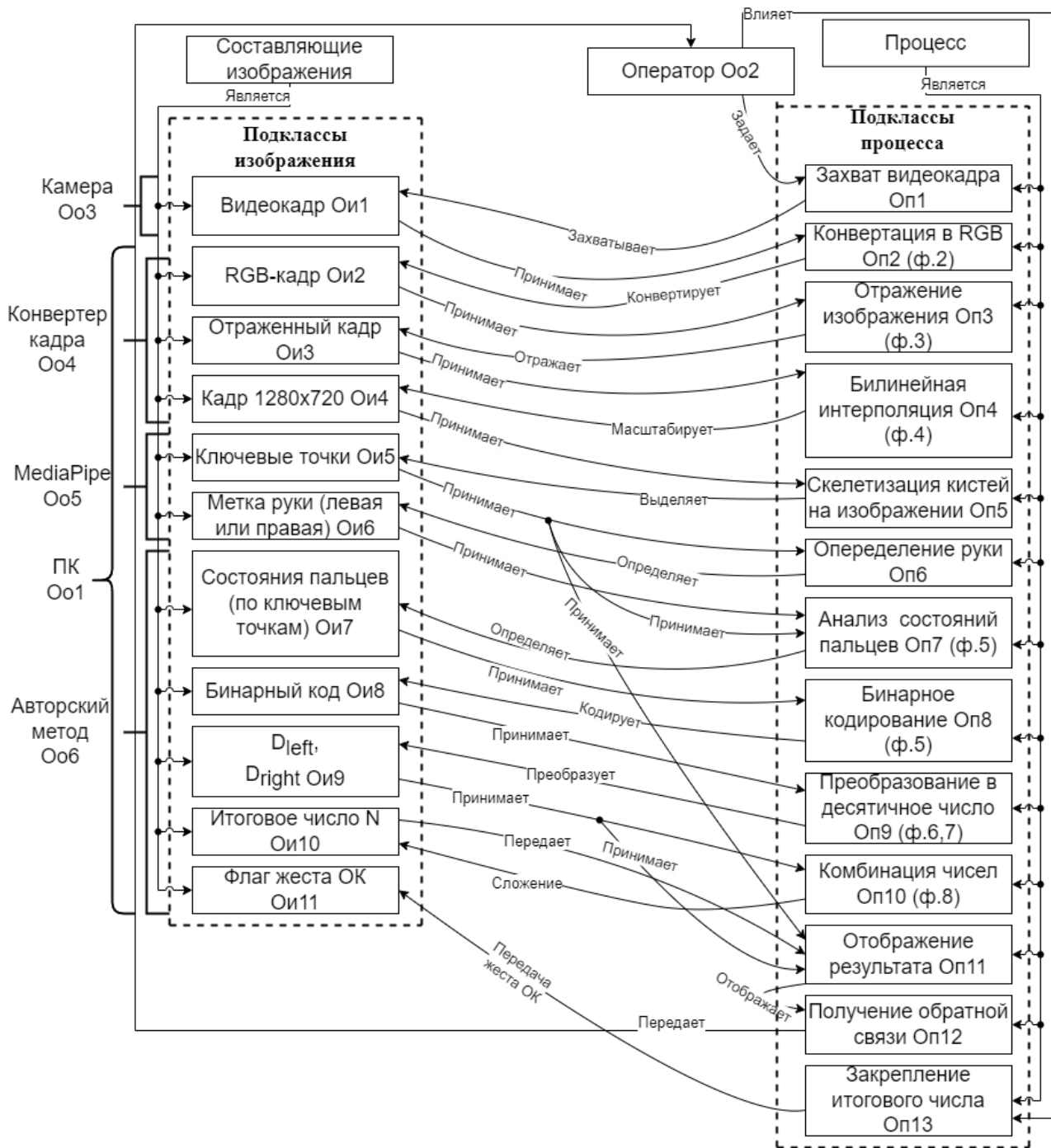


Рис. 1. Онтологическая модель распознавания жестов рук оператора.

Табл. 1. Онтология процессов, сигналов, объектов системы распознавания жестов рук оператора.

№	Процесс	Описание процесса	Атрибуты		Связанные объекты
			Входные сигналы	Выходные сигналы	
1	Захват видеокadra (ф. 2)	Получение кадра с камеры	–	Видеокادر	Камера, ПК
2	Конвертация в RGB (ф. 3)	Преобразование цветового пространства YUYV в RGB	Видеокادر	RGB-кадр	Конвертер кадра
3	Отражение изображения (ф. 3)	Горизонтальное отзеркаливание положения рук	RGB-кадр	Отраженный кадр	Конвертер кадра
4	Билинейная интерполяция (ф. 4)	Приведение кадра к разрешению 1280 x 720	RGB-кадр	Масштабированный кадр	Конвертер кадра
5	Скелетизация кистей (ф. 5)	Определение наличия рук и их ключевых точек	Масштабированный кадр	Ключевые точки	MediaPipe
6	Определение руки	Классификация обнаруженной руки (левая/правая)	Ключевые точки	Метка руки	MediaPipe
7	Анализ состояний пальцев	Сравнение координат кончиков и оснований пальцев	Ключевые точки, метка руки	Состояния пальцев	Авторский метод
8	Бинарное кодирование (ф. 6)	Объединение состояний пальцев в бинарную строку	Состояния пальцев	Бинарный код	Авторский метод
9	Преобразование в десятичное число (ф. 6, 7)	Конвертация бинарного кода в десятичное число (0–9)	Бинарный код	D_{left} или D_{right}	Авторский метод
10	Комбинация чисел (ф. 8)	Вычисление итогового числа N_{final}	$D_{\text{left}}, D_{\text{right}}$	Итоговое число N_{final}	Авторский метод
11	Отображение Результата (ф. 9)	Вывод итогового числа в графическом интерфейсе	Итоговое число N_{final}	отображается	Графический интерфейс
12	Получение обратной связи	Прием корректирующих данных от оператора	Действия оператора	Новые жесты	Графический интерфейс, Оператор

13	Подтверждение итогового числа	Использование жеста «ОК» для фиксации N_{final}	Жест опера- тора	Флаг жеста «ОК»	Графиче- ский интер- фейс, Опе- ратор
----	----------------------------------	--	---------------------	--------------------	--

Примечание. Обозначения ф. 1, ф. 2, ..., ф. 9 указывают на соответствующие формулы, представленные ниже.

На уровне физических объектов формируются визуальные сигналы, соответствующие реальному видеопотоку. Камера захватывает видеоряд, при этом формируется видеокадр, который поступает в вычислительную среду ПК. Далее система переходит на уровень цифровых объектов, реализованных внутри программного кода в ПК. Конвертер кадра выполняет преобразование цветового пространства изображения из YUV в RGB, отражает его, далее с помощью масштабирования кадр приводится к разрешению 1280 x 720. После предобработки кадра библиотека MediaPipe выполняет детекцию рук на нем, формируя координаты 21 ключевой точек (кончики и суставы пальцев), и определение руки (левая или правая), возвращая метку руки (D_{left} или D_{right}).

Наш метод кодирует состояния пальцев по их ключевым точкам в бинарный код. Далее двоичное значение каждой из рук конвертируется в десятичное число. На выходе онтологической модели формируется десятичное число за счет суммирования значений бинарных кодов левой и правой рук. При необходимости оператор подает сигнал обратной связи для калибровки параметров системы или фиксации итогового числа (рис. 1) [16, 17].

Таким образом, онтологическая модель распознавания жестов рук оператора предназначена для преобразования кадра, полученного с камеры, в итоговое десятичное значение при анализе расположения ключевых точек скелетизации. При этом каждая формула реализует конкретный онтологический процесс и использует сигналы, источники которых заданы в онтологии (табл. 1). Для реализации онтологической модели необходимо выполнить следующую последовательность шагов.

Шаг 1. Предобработка изображения

Захват видеопотока выполняется камерой и обеспечивает получение последовательности кадров. Выходной сигнал (видеокадр) представляет собой

изображение в формате YUYV, предоставляемом камерой, и преобразуется в RGB по формуле

$$I_{\text{RGB}}(x, y, c) = I_{\text{origin}}(x, y, 2 - c), \quad (2)$$

где (x, y) – координаты пикселя в выходном кадре ($0 \leq x < W$, $0 \leq y < H$), c – индекс цветового канала ($c = 0$ для красного, 1 – для зеленого, 2 – для синего цветов), I_{RGB} – результирующий RGB-кадр.

Таким образом, исходный синий канал ($c = 2$) становится красным, а красный ($c = 0$) – синим. Далее происходит горизонтальное «отзеркаливание» кадра. Это необходимо для корректной ориентации изображения, так как при фронтальной камере изображение отображается зеркально относительно реального положения рук оператора. Преобразование задается формулой

$$I_{\text{reflect}}(x, y) = I_{\text{RGB}}(W - 1 - x, y), \quad (3)$$

где I_{reflect} – отраженный кадр, W – ширина кадра в пикселях, x и y – целочисленные координаты пикселя ($0 \leq x < W$, $0 \leq y < H$, где H – высота кадра).

Масштабирование изменяет размер кадра до фиксированного разрешения (1280×720 пикселей) с сохранением пропорций. При использовании билинейной интерполяции преобразование задается формулой

$$I_{\text{scale}}(x', y') = \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} I_{\text{RGB}}(i, j) \left(0, 1 - \left| i - y' \frac{H}{H'} \right| \right) \max \left(0, 1 - \left| j - x' \frac{W}{W'} \right| \right), \quad (4)$$

где H, W – высота и ширина исходного RGB-кадра, H', W' – целевая высота и ширина масштабированного кадра, (x', y') – координаты пикселя в выходном кадре ($0 \leq x' < W'$, $0 \leq y' < H'$), $I_{\text{RGB}}(i, j)$ – значение пикселя исходного кадра в строке i и столбце j .

Для областей, выходящих за пределы исходного изображения после масштабирования с сохранением пропорций, значение пикселей принимается равным нулю (черный цвет).

Шаг 2. Скелетизация рук и классификация, как левой или правой

Детекция рук на изображении реализуется библиотекой MediaPipe Hands. На вход подается масштабированный кадр I_{scale} , а на выходе формируются ключевые точки, представляющие собой массив трехмерных координат (x, y, z) для 21 ключевой точек каждой обнаруженной руки по формуле

$$O_{m,n,c} = \sum_{i=0}^{k-1} \sum_{j=0}^{k-1} \sum_{d=0}^{D-1} I_{scaled}(m+i, n+j, d) F_{i,j,d,c} + b_c, \quad (5)$$

где $O_{m,n,c}$ – значение выходного признака (карты признаков) в позиции (m, n) для канала c ; k – размер ядра свертки; D – число входных каналов (3 – для RGB); $F_{i,j,d,c}$ – веса фильтра; b_c – аддитивное смещение (bias).

Координаты распознанных ключевых точек нормированы относительно размеров кадра и глубины (рис. 2).

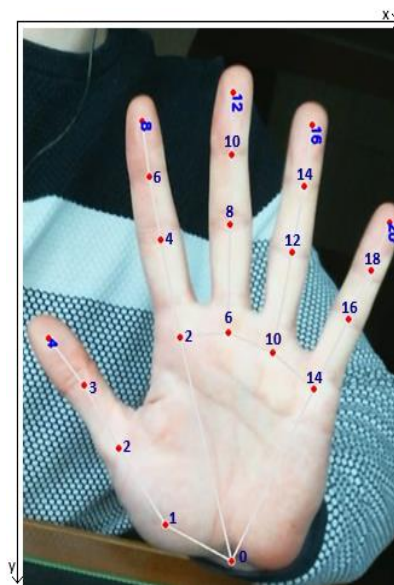


Рис. 2. Распознанные ключевые точки.

Далее выполняется анализ полученных ключевых точек и классификация обнаруженной руки, левой или правой. Эта информация необходима для корректной интерпретации жестов, поскольку для левой и правой рук правила определения поднятых пальцев различаются (для большого пальца знак сравнения по оси x инвертируется). Таким образом, выходной сигнал (метка руки) принимает значения “left” или “right”.

Шаг 3. Определение состояний пальцев

Когнитивная интерпретация жестов оператора основана на анализе взаимного расположения ключевых точек кисти, полученных с камеры. Состояние каждого пальца («поднят» – 1, «согнут» – 0) определяется по правилам, приведенным в табл. 2, где в качестве опорной точки используется сустав, идущий через одну позицию за кончиком пальца (индекс кончика i , индекс опоры $i - 2$). Для всех пальцев, кроме большого, сравнение выполняется по координате ординат y (вертикальное движение), для большого пальца – по координате абсцисс x (горизонтальное движение). Для левой руки правило большого пальца инвертируется в соответствии с формулой (3).

Табл. 2. Онтология процессов, сигналов, объектов системы распознавания жестов рук оператора.

Палец	Индексы точек (кончик / опора)	Правило для правой руки (1 – поднят)	Правило для левой руки (1 – поднят)
Большой	4 / 3	$x_{\text{кончик}} > x_{\text{опора}}$	$x_{\text{кончик}} < x_{\text{опора}}$
Указательный	8 / 6	$y_{\text{кончик}} < y_{\text{опора}}$	$y_{\text{кончик}} < y_{\text{опора}}$
Средний	12 / 10	$y_{\text{кончик}} < y_{\text{опора}}$	$y_{\text{кончик}} < y_{\text{опора}}$
Безымянный	16 / 14	$y_{\text{кончик}} < y_{\text{опора}}$	$y_{\text{кончик}} < y_{\text{опора}}$
Мизинец	20 / 18	$y_{\text{кончик}} < y_{\text{опора}}$	$y_{\text{кончик}} < y_{\text{опора}}$

Бинарный код жеста, соответствующий вектору состояний пальцев для одной руки S , определяется выражением

$$B = (S_{\text{thumb}}, S_{\text{index}}, S_{\text{middle}}, S_{\text{ring}}, S_{\text{pinky}}), \quad (6)$$

где S_{thumb} , S_{index} , S_{middle} , S_{ring} , S_{pinky} – бинарные состояния большого, указательного, среднего, безымянного пальцев и мизинца соответственно (1 – поднят, 0 – согнут).

Например, если подняты только безымянный палец и мизинец, а остальные согнуты, то для правой руки получим $B = (0, 0, 0, 1, 1)$, что соответствует бинарному коду 00011 и числу 3 в десятичной кодировке.

Шаг 4. Бинарно-десятичное преобразование числа

Преобразование бинарного кода B в десятичное число выполняется по формуле

$$D_{\text{raw}} = \sum_{j=0}^4 B_j 2^{(4-j)}, \quad (7)$$

где B_j – состояние j -го пальца из вектора B ; $j = 0$ соответствует большому пальцу ($2^4 = 16$), $j = 1$ – указательному (вес $2^3 = 8$), $j = 2$ – среднему ($2^2 = 4$), $j = 3$ – безымянному ($2^1 = 2$), $j = 4$ – мизинцу ($2^0 = 1$).

Полученное десятичное значение D находится в диапазоне от 0 до 31, $D_{\text{raw}} \in [0; 31]$. Для приведения к диапазону десятичных цифр от 0 до 9 используется ограничение

$$D = (D_{\text{raw}}, 9). \quad (8)$$

С учетом формул (6)÷(8) итоговое десятичное двузначное число N_{final} вычисляется по формуле

$$N_{\text{final}} = 10D_{\text{left}} + D_{\text{right}}, \quad (9)$$

Результат вычислений выводится на графический интерфейс, а также передается на систему управления ПИД-регулятором [3].

На первом шаге инициализации принимается $D_{\text{left}} = D_{\text{right}} = 0$. После каждого кадра интерпретация жестов, сформированных левой и правой руками (в соответствии с актуальными состояниями их пальцев), обновляется, обеспечивая работу системы в режиме реального времени и/или близком к нему.

Шаг 5. Человеко-машинный интерфейс

Итоговое десятичное число (N_{final}), определяемое по формуле (9), посредством графического интерфейса передаётся на вход ПИД-регулятора, устанавливая в законе управления требуемое значение (уставку), относительно которого вычисляется ошибка регулирования как разность между уставкой и текущим измеренным значением регулируемой величины. Интерфейс также отображает текущие состояния пальцев каждой руки, наглядно демонстрирует ске-

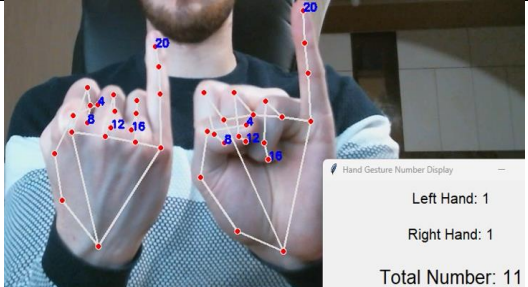
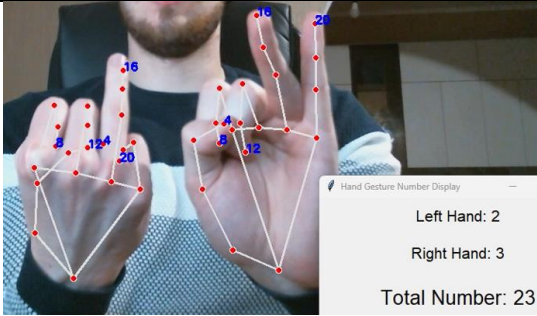
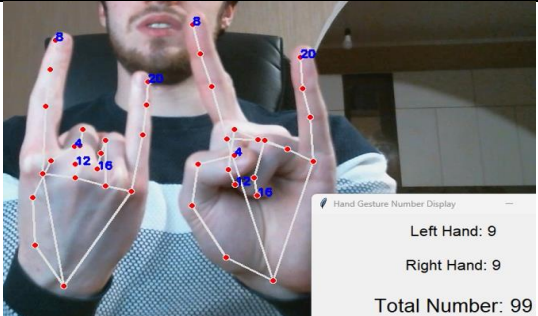
летизацию кистей и выводит состояние флага жеста «ОК». Использование данного жеста позволяет понятным человеку жестом зафиксировать/освободить значение итогового числа N_{final} .

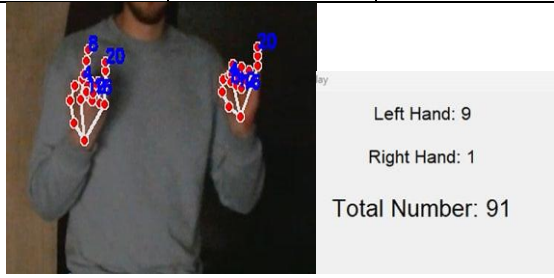
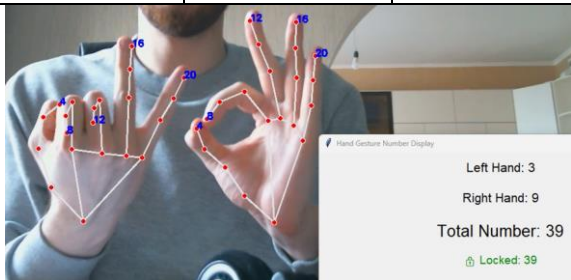
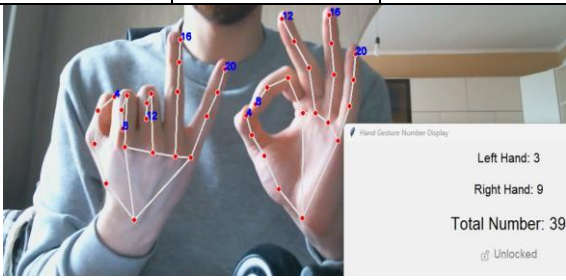
ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ ОНТОЛОГИЧЕСКОЙ МОДЕЛИ РАСПОЗНАВАНИЯ ЖЕСТОВ РУК ОПЕРАТОРА

Экспериментальные исследования выполнялись на персональном компьютере под управлением операционной системы Windows 11 с процессором Intel(R) Xeon(R) CPU E5-2667 v4 @ 3,20 ГГц, оперативной памятью 48 Гб DDR4 и веб-камерой Logitech C615 (разрешение 1920×1080 FHD, угол обзора 78°, частота кадров 30 кадр./с).

В табл. 3 представлено соответствие между жестами, заданными оператором, и получаемыми выходными числами, определяемыми по формуле (9). Данная оценка осуществлялась визуально.

Табл. 3. Формирование числовых значений по жестам оператора.

№ эксп.	Жест (левая рука)	Жест (правая рука)	Номер	Жест (левая рука)	Жест (правая рука)	Номер
1	0	0	0	1	1	11
						
2	2	3	23	9	9	99
						

3	3	1	31	3	3	33
						
4	Жест (левая рука)	Жест (правая рука)	Номер	Жест (левая рука)	Жест (правая рука)	Номер
	8	1	81	9	1	91
						
5	3	9, «OK»	39 (Locked)	3	9, «OK»	39 (Unlocked)
						

В ходе исследования было выполнено пять экспериментов. Эксперименты 1 и 2 проводились в нормальных условиях освещения (лампа 80 Вт, освещенность около 300 люкс) при расстоянии от оператора до камеры 1.2 м. В ходе процесса оценивалось корректное формирование десятичных чисел для комбинаций жестов оператора. В общей сложности было выполнено 100 распознаваний для каждой пары чисел от 0 до 99. Система продемонстрировала устойчивое распознавание жестов с точностью 98%.

В эксперименте 3, проведенном в сложных условиях низкой освещенности (50 лк) и при увеличенном расстоянии до камеры (2 м), система сохранила точность 94%, что демонстрирует ее робастность.

В эксперименте 4 расстояние от оператора до камеры увеличивалось до тех пор, пока точность не опускалась ниже приемлемого порога. В хороших условиях освещения (лампа 80 Вт) граничное расстояние составило 4.2 м при точности около 70%. При неблагоприятном освещении (два LCD-монитора диагональю 27 дюймов: один яркостью 300 кд/м² с белым спектром изображения, второй яркостью 250 кд/м² с серым) максимальная дальность корректного распознавания снизилась до 2.8 м. Таким образом, качество освещения является важным фактором, определяющим рабочий диапазон дальности системы.

В эксперименте 5 проводилась проверка когнитивного восприятия системой жеста «ОК» для фиксации/освобождения итогового десятичного значения.

В каждом сценарии было проведено не менее 30 независимых испытаний для каждого жеста. Жестовый набор включал все комбинации поднятых пальцев для левой и правой рук (всего 100 возможных двузначных чисел от 0 до 99). Для каждого жеста регистрировались результаты распознавания.

Метрика точности

Для расчета показателей эффективности ключевой метрикой была выбрана точность распознавания жестов, которая вычислялась по формуле

$$\text{Точность} = \frac{\text{ПР}}{\text{ПР} + \text{ЛС} + \text{ОС}} \cdot 100\%, \quad (10)$$

где ПР (Правильное Распознавание) – количество правильно распознанных целевых жестов, ЛС (Ложное Распознавание) – количество ложных срабатываний (распознан жест, который не показывался), ОС (Отсутствие Срабатывания) – количество пропущенных целевых жестов.

Анализ экспериментальных результатов

На основе полученных данных и с учетом формулы (10) были рассчитаны ключевые показатели эффективности в сравнении с базовыми методами (OpenCV на основе контурного анализа и TensorFlow с предобученной моделью). Результаты представлены в табл. 4.

Табл. 4. Сравнительные характеристики систем распознавания жестов.

Параметр	ОМСРЖО	OpenCV	TensorFlow	Абсолютное улучшение ОМСРЖО vs TensorFlow
Точность (норм. усл.), %	98.0	88.0	95.0	+3.0
Точность (шум), %	95.5	80.0	88.0	+7.5
Точность (плохой свет), %	94.0	78.0	85.0	+9.0
Время отклика (норм. усл.), с	0.50	0.60	0.85	−0.35
Устойчивость к шуму (норм. усл.), %	−2.5	−8.0	−7.0	-

Система достигла средней точности распознавания 98 % в нормальных условиях. Количество ошибок распознавания росло в основном при расположении рук слишком близко к камере (< 0.5 м) или слишком далеко от нее (> 4.2 м), а также при недостаточном освещении (< 100 лк). Среднее время отклика составило 0.5 с, что позволяет использовать систему в реальном времени. Таким образом, результаты эксперимента подтверждают преимущество ОМСРЖО над OpenCV и TensorFlow, которое состоит в увеличении точности распознавания заданных жестов.

ЗАКЛЮЧЕНИЕ

Разработана онтологическая модель преобразования жестов рук оператора в десятичный цифровой сигнал. В ее основе лежит когнитивный принцип сравнения относительного положения кончика пальца и его основания, что имитирует естественное для человека восприятие жеста. Кроме того, онтологическая модель формализует состав системы, типы сигналов и процессы, а также реализует вычислительную интерпретацию каждого из этих процессов. Алгоритмы определения состояний пальцев, бинарно-десятичного преобразования и комбинирования чисел от двух рук соответствуют конкретным онтологическим

процессам и используют строго определенные онтологические сигналы. Это обеспечивает сквозную прослеживаемость от физического объекта до получения итогового числа передающегося, например, на исполнительные устройства мехатронных комплексов.

Экспериментальные результаты подтверждают работоспособность предложенного подхода. Система достигла точности распознавания 98% в нормальных условиях и 94% при наличии шума и низкой освещенности. По точности и времени отклика полученные показатели превосходят решения на базе OpenCV и TensorFlow. Результаты свидетельствуют о перспективности онтологического подхода для проектирования и формализации сложных систем распознавания образов в реальном времени.

Вместе с тем реализованная онтологическая модель имеет ряд ограничений, определяющих направления будущих исследований. Алгоритм скелетизации MediaPipe испытывает затруднения при корректном определении положения большого пальца, поскольку его движение происходит в горизонтальной плоскости, а не в вертикальной, что в некоторых случаях приводит к ошибкам кодирования расположения ключевых точек на кисти руки. Шум видеопотока и недостаточная освещенность (50 лк и менее) снижают точность распознавания. Для устранения перечисленных ограничений планируется усовершенствование базы нечетко-логических правил в процессе обработки распознанных ключевых точек.

Благодарности

Работа выполнена при поддержке Министерства науки и высшего образования в рамках выполнения работ по Государственному заданию № 075-03-2026-489.

СПИСОК ЛИТЕРАТУРЫ

1. Кумашев Э., Аминов Н.С. Распознавание жестов в умном доме // XII Конгресс молодых ученых: сборник научных трудов. СПб.: Университет ИТМО, 2023. С. 201–211.
2. Ефремов А.В., Муратов Е.Р., Никифоров М.Б., Хутев А.Н. Распознавание жестов в бесконтактном управлении робототехническими системами //

Биотехнические, медицинские и экологические системы, измерительные устройства и робототехнические комплексы – Биомедсистемы-2022: сборник трудов XXXV Всероссийской научно-технической конференции студентов, молодых ученых и специалистов. Рязань: ИП Коняхин А.В., 2022. С. 446–448.

3. *Бобырь М.В., Милостная Н.А., Ноливос К.А.* Комбинация нечетко-цифрового фильтра и ПИД регулятора в задаче управления термоэлементом // Мехатроника, автоматизация, управление. 2022. Том 23. С. 473-480.

<https://doi.org/10.17587/mau.23.473-480>

4. *Саттарова В.В., Хачумов М.В.* Сравнительный анализ методов машинного обучения в задаче распознавания жестов рук // Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологических систем: материалы Всероссийской конференции с международным участием. Москва: РУДН им. П. Лумумбы, 2024. С. 221–226.

5. *Куликова Э.В.* Алгоритм распознавания жестов русского жестового языка на видео // Заметки по информатике и математике: сборник научных статей. Вып. 15. Ярославль: ЯрГУ им. П.Г. Демидова, 2023. С. 41–50.

6. *Althubiti A.H., Algethami H.* Gesture Recognition Using Deep Learning Algorithms: A Comparative Study // International Journal of Advanced Computer Science and Applications. 2024. Vol. 15, No. 6.

7. *Bhujbal P.S., Kotmale M., Kadam N., Magar M., Barakade K.* Sign Observation and Understanding Language (SOUL) // International Journal of Engineering Applied Sciences and Technology. 2024. Vol. 09, Issue 5. P. 135–139.

<https://doi.org/10.33564/ijeast.2024.v09i05.016>

8. *Dhamodaran S., Phukan P.P., Singh M., Nandakum Sh.* Implementation of Hand Gesture Recognition Using OpenCV // International Journal of Research Publication and Reviews. 2024. Vol. 5, No. 5. P. 8039–8050.

<https://doi.org/10.55248/gengpi.5.0524.1324>

9. *Han Lin H.Ye., Murli N.* BIM Sign Language Translator Using Machine Learning (TensorFlow) // Journal of Soft Computing and Data Mining. 2022. Vol. 3, No. 1. <https://doi.org/10.30880/jscdm.2022.03.01.007>

10. *Muhammad M.A., Martinus M., Nurhartanto A., Mulyani Ye., Djausal G.P., Achmad D., Ferbangkara S.* Development of Lampung Script Characters

Recognition Model Using TensorFlow // International Journal of Electronics and Communications Systems. 2023. Vol. 3, No. 2. P. 97.

<https://doi.org/10.24042/ijecs.v3i2.19878>

11. Controlling Websites Through Hand Gestures Using TensorFlow JS and Browser Extension // International Research Journal of Modernization in Engineering Technology and Science. 2023. <https://doi.org/10.56726/irjmets34907>

12. Kocakulak M., Acir N. A Contactless Palmprint Imaging System Design Using MediaPipe Hands // Journal of Innovative Science and Engineering (JISE). 2022. <https://doi.org/10.38088/jise.1142634>

13. Song R., Hong Yu., Kwak N. User Interface Using Hand Gesture Recognition Based on MediaPipe Hands Model // Journal of Korea Multimedia Society. 2023. Vol. 26, No. 2. P. 103–115. <https://doi.org/10.9717/kmms.2023.26.2.103>

14. Pati R.N., Vallabhaneni N., P P., Shekhawat Ya., Paria S. Real-Time Computer Vision Based Hand Gesture Recognition // International Research Journal on Advanced Science Hub. 2023. Vol. 5, Issue 05S. P. 382–391. <https://doi.org/10.47392/irjash.2023.s052>

15. Grebeshkov A., Shebalov R., Gorshkov S., Mushtak O. Ontological modeling of enterprises: technologies and methods // Book, TriniData LLC & Ural Federal University. 2019. P. 104–116. ISBN: 978-5-7996-2580-1.

16. Бобырь М.В., Асеев А.А. Программа управления жестами рук с вводом числовых команд на основе библиотеки MediaPipe. Свидетельство о государственной регистрации программы для ЭВМ № 2026616114. Дата регистрации: 04.03.2026.

17. Бобырь М.В., Асеев А.А. Программа для анализа жестов рук и выделения ключевых точек на основе библиотек MediaPipe и TensorFlow. Свидетельство о государственной регистрации программы для ЭВМ № 2026616251. Дата регистрации: 05.03.2026.

ONTOLOGICAL MODEL FOR OPERATOR HAND GESTURE RECOGNITION

M. V. Bobyr¹ [0000-0002-5400-6817], A. A. Aseev² [0009-0007-8271-7660]

^{1, 2}*Southwest State University, Kursk, Russia*

¹maxboby@gmail.com, ²asseeff.artem@gmail.com

Abstract

The article presents an ontological model for operator hand gesture recognition based on hand skeletonization using the MediaPipe Hands library and an original method for cognitive transformation of finger states from skeleton keypoints into binary code with its subsequent conversion into a digital control signal. Improvement in the technical and methodological characteristics of the system is achieved through explicit definition of cause-and-effect relationships between objects and processes within the ontology, as well as the application of clear binary encoding rules. The mathematical model of the system includes the following stages: video frame preprocessing, hand skeletonization, coordinate analysis of keypoints, binary encoding of finger states, and generation of the final decimal control signal. The developed gesture recognition interface ensures an accuracy of 98% under normal conditions and no less than 94% under noisy and low-light conditions (50 lux or less), with an average response time of 0.5 seconds, outperforming baseline methods based on widely used libraries such as OpenCV and TensorFlow.

Keywords: *gesture recognition, ontology, MediaPipe, skeletonization algorithm, binary code, graphical user interface, cognitivity.*

Acknowledgements

This work was supported by the Ministry of Science and Higher Education under State Contract No. 075-03-2026-489.

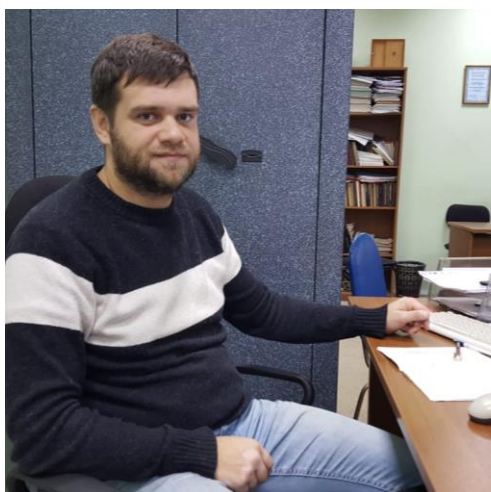
REFERENCES

1. Kumashev E., Aminov N.S. Gesture Recognition in a Smart Home // XII Congress of Young Scientists: Collection of Scientific Papers. St. Petersburg: ITMO University, 2023. P. 201–211.

2. *Efremov A.V., Muratov E.R., Nikiforov M.B., Khutev A.N.* Gesture Recognition in Contactless Control of Robotic Systems // Biotechnical, Medical and Environmental Systems, Measuring Devices and Robotic Complexes – Biomedsystems-2022: Proceedings of the XXXV All-Russian Scientific and Technical Conference of Students, Young Scientists and Specialists. Ryazan: IP Konyakhin A.V., 2022. P. 446–448.
3. *Bobyr M., Milostnaya N., Nolivos C.* PID Regulator and Fuzzy-Digital filter combination in a thermoelement control task // Mechatronics, Automation, Control. 2022. Vol. 23. P. 473–480. <https://doi.org/10.17587/mau.23.473-480>
4. *Sattarova V.V., Khachumov M.V.* Comparative Analysis of Machine Learning Methods in the Task of Hand Gesture Recognition // Information and Telecommunication Technologies and Mathematical Modeling of High-Tech Systems: Proceedings of the All-Russian Conference with International Participation. Moscow: RUDN University named after P. Lumumba, 2024. P. 221–226.
5. *Kulikova E.V.* Algorithm for Recognizing Russian Sign Language Gestures in Video // Notes on Informatics and Mathematics: Collection of Scientific Articles. Issue 15. Yaroslavl: Yaroslavl State University named after P.G. Demidov, 2023. P. 41–50.
6. *Althubiti A.H., Algethami H.* Gesture Recognition Using Deep Learning Algorithms: A Comparative Study // International Journal of Advanced Computer Science and Applications. 2024. Vol. 15, No. 6.
7. *Bhujbal P.S., Kotmale M., Kadam N., Magar M., Barakade K.* Sign Observation and Understanding Language (SOUL) // International Journal of Engineering Applied Sciences and Technology. 2024. Vol. 09, Issue 5. P. 135–139. <https://doi.org/10.33564/ijeast.2024.v09i05.016>
8. *Dhamodaran S., Phukan P.P., Singh M., Nandakum Sh.* Implementation of Hand Gesture Recognition Using OpenCV // International Journal of Research Publication and Reviews. 2024. Vol. 5, No. 5. P. 8039–8050. <https://doi.org/10.55248/gengpi.5.0524.1324>
9. *Han Lin H.Ye., Murli N.* BIM Sign Language Translator Using Machine Learning (TensorFlow) // Journal of Soft Computing and Data Mining. 2022. Vol. 3, No. 1. <https://doi.org/10.30880/jscdm.2022.03.01.007>

10. *Muhammad M.A., Martinus M., Nurhartanto A., Mulyani Ye., Djausal G.P., Achmad D., Ferbangkara S.* Development of Lampung Script Characters Recognition Model Using TensorFlow // International Journal of Electronics and Communications Systems. 2023. Vol. 3, No. 2. P. 97.
<https://doi.org/10.24042/ijecs.v3i2.19878>
11. *Outanoute M.*, Controlling Websites Through Hand Gestures Using TensorFlow JS and Browser Extension // International Research Journal of Modernization in Engineering Technology and Science. 2023.
<https://doi.org/10.56726/irjmets34907>
12. *Kocakulak M., Acir N.* A Contactless Palmprint Imaging System Design Using MediaPipe Hands // Journal of Innovative Science and Engineering (JISE). 2022.
<https://doi.org/10.38088/jise.1142634>
13. *Song R., Hong Yu., Kwak N.* User Interface Using Hand Gesture Recognition Based on MediaPipe Hands Model // Journal of Korea Multimedia Society. 2023. Vol. 26, No. 2. P. 103–115. <https://doi.org/10.9717/kmms.2023.26.2.103>
14. *Pati R.N., Vallabhaneni N., P P., Shekhawat Ya., Paria S.* Real-Time Computer Vision Based Hand Gesture Recognition // International Research Journal on Advanced Science Hub. 2023. Vol. 5, Issue 05S. P. 382–391.
<https://doi.org/10.47392/irjash.2023.s052>
15. *Grebeshev A., Shebalov R., Gorshkov S., Mushtak O.* Ontological modeling of enterprises: technologies and methods // Book, TriniData LLC & Ural Federal University. 2019. P. 104–116. ISBN: 978-5-7996-2580-1.
16. *Boby M.V., Aseev A.A.* Program for Hand Gesture Control with Numerical Command Input Based on the MediaPipe Library. Certificate of State Registration of Computer Program No. 2026616114. Registration date: March 4, 2026.
17. *Boby M.V., Aseev A.A.* Program for Hand Gesture Analysis and Key Point Extraction Based on MediaPipe and TensorFlow Libraries. Certificate of State Registration of Computer Program No. 2026616251. Registration date: March 5, 2026.

СВЕДЕНИЯ ОБ АВТОРАХ



БОБЫРЬ Максим Владимирович – 1978 года рождения, учился в Курском государственном техническом университете (ныне – Юго-Западный государственный университет). В 2012 году защитил диссертацию на соискание доктора технических наук по специальности 05.13.06 «Автоматизация и управление технологическими процессами и производствами». В настоящее время работает профессором на кафедре программной инженерии. Является председателем диссертационного совета по специальности 5.12.4 «Когнитивное моделирование».

Область научных интересов: адаптивные нейронечеткие системы вывода, цифровая обработка изображений и распознавание образов, машинное обучение, стереозрение.

Maksim Vladimirovich BOBYR – born in 1978, studied at Kursk State Technical University (now Southwest State University). In 2012, he defended his dissertation for the degree of Doctor of Technical Sciences in specialty 05.13.06 Automation and Control of Technological Processes and Production. Currently, he works as a professor at the Department of Software Engineering. He is the chairman of the dissertation council for specialty 5.12.4 Cognitive Modeling.

Research interests: adaptive neuro-fuzzy inference systems, digital image processing and pattern recognition, machine learning, stereo vision.

email: maxboby@gmail.com

ORCID: 0000-0002-5400-6817



АСЕЕВ Артем Андреевич – 1999 года рождения, учится на 3 курсе аспирантуры Юго-Западного государственного университета на кафедре программной инженерии по специальности 2.3.3 Автоматизация и управление технологическими процессами и производствами. Область научных интересов: адаптивные нейро-нечеткие системы вывода, распознавание образов.

Artem Andreevich ASEEV – born in 1999, a 3th-year postgraduate student at the Southwest State University, Department of Software Engineering, specialty 2.3.3 Automation and Control of Technological Processes and Production. Research interests: adaptive neuro-fuzzy inference systems, pattern recognition.

email: asseeff.artem@gmail.com

ORCID: 0009-0007-8271-7660

Материал поступил в редакцию 22 апреля 2026 года