

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Кибернетика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

Елшібеков Елжас Рашидұлы

Адамдарды тану арқылы электрондық қатысу жүйесін әзірлеу

ДИПЛОМДЫҚ ЖҰМЫС

5B070300 – «Ақпараттық жүйелер» мамандығы

Алматы 2020

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Кибернетика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

ҚОРҒАУҒА ЖІБЕРІЛДІ
КАӨЖС кафедрасы меңгерушісі
техн. ғыл. канд., ассистент-
профессор
_____ Н.А. Сейлова
«__» _____ 2020 ж.

ДИПЛОМДЫҚ ЖҰМЫС

Тақырыбы: «Адамдарды тану арқылы электрондық қатысу жүйесін әзірлеу»

5B070300 – «Ақпараттық жүйелер» мамандығы

Орындаған :

Елшібеков Е.Р.

Пікір беруші :

Ғылыми жетекші

_____ Н. Ж. Таубалды

«__» _____ 2020 ж.

«__» _____ 2020 ж.

Алматы 2020

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Кибернетика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

5B070300 – Ақпараттық жүйелер

БЕКІТЕМІН

КАӨЖС кафедрасы меңгерушісі
техн. ғыл. канд., ассистент-
профессор

_____ Н.А. Сейлова
«__» _____ 2020 ж.

**Дипломдық жұмысты орындауға
ТАПСЫРМА**

Білім алушы Елшібеков Елжас Рашидұлы

Жобаның тақырыбы: «Адамдарды тану арқылы электрондық қатысу жүйесін
өзірлеу»

Университеттің «27» қаңтар 2020 жылғы ғылыми кеңесінің № 762-б шешімімен
бекітілген.

Орындалған жұмыстың өткізу мерзімі «__» _____ 2020 ж.

Дипломдық жұмыстың бастапқы мәліметтері: берілген тақырып бойынша
әдебиеттерге шолу кезінде жиналған мәліметтер, теориялық материалдар жинау.

Дипломдық жұмыста қарастырылатын мәселелер тізімі:

- а) тақырып аясын талдау;
- ә) несиелік калькулятор үшін есептеу әдістерін таңдау;
- б) бағдарламалық қамтамасыздандыруды жасау.

Графикалық материалдардың тізімі (міндетті түрде қажет сызбалар көрсетілген):
жұмыстың ____ слайдтан тұратын презентациясы көрсетіледі.

Негізгі әдебиет 14 атау.

Дипломдық жұмысты даярлау
КЕСТЕСІ

Бөлім атаулары, қарастырылатын мәселелер тізімі	Ғылыми жетекшіге, кеңесшілерге өткізу мерзімі	Ескерту
Тақырып аясын талдау	20.01.2020 – 07.02.2020	
Адамдарды тану арқылы электрондық қатысу жүйесін әзірлеу	08.02.2020 – 10.03.2020	
Бағдарламалық қамтамасыздандыруды жасау	11.03.2020 – 28.04.2020	

Дипломдық жұмыс бөлімдерінің кеңесшілері мен
норма бақылаушының аяқталған жұмысқа қойған
қолтаңбалары

Бөлімдердің атауы	Кеңесшілер (аты-жөні, тегі, ғылыми дәрежесі, атағы)	Қол қойылған мерзімі	Қолы
БҚ жасау	Н. Ж. Таубалды техн.ғыл.магистрі, лектор		
Норма бақылаушы	А. Н. Дуйсенбаева техн.ғыл.магистр., лектор		

Ғылыми жетекшісі _____ Н. Ж. Таубалды

Тапсырманы орындауға қабылдаған білім алушы _____ Е. Р. Елшібеков

Күні

«27» Қаңтар 2020 ж.

REPORT DATE: 2020-05-28 09:58:41

TITLE:

1.docx Адамдарды тану арқылы электрондық қатысу жүйесін әзірлеу

AUTHOR(S):

Елшібеков Е.Р.-Повторно

PROMOTER:

Нұрсұлтан Таубалды

ORGANIZATIONAL UNIT:

ИКиИТ

FILE UPLOAD DATE:

2020-05-28 09:56:18

NUMBER OF VERIFICATIONS:

1

SKIPPED URL ADDRESSES:

Record of similarities

Please note that high coefficient values do not automatically mean plagiarism. The report must be analyzed by an authorized person.



Alerts

In this section, you can find information regarding text modifications that may aim at temper with the analysis results. Invisible to the person evaluating the content of the document on a printout or in a file, they influence the phrases compared during text analysis (by causing intended misspellings) to conceal borrowings as well as to falsify values in the Similarity Report. It should be assessed whether the modifications are intentional or not.

Characters from another alphabet	48	show in the text
number of characters from other alphabets which may imitate letters from the alphabet relevant to the document, causing misspellings in the text, please verify their validity		
Spreads	0	show in the text
number of increased distances between letters - please verify whether they imitate spaces, indicating as merging the words in the Report		
Micro spaces	0	show in the text
number of spaces with zero length - please verify whether they are placed inside words and cause word division in the text		
White characters	0	show in the text
number of characters with a white font color - please verify whether they are used instead of spaces, causing merge of the words (in the Report the color of the letters is changed to black in order to show them)		

Active lists of similarities

Scroll the list and analyze especially the fragments that exceed the SC 2 (marked in bold). Use the link "Mark fragment" and see if they are short phrases scattered in the document (coincidental similarities), numerous short phrases near each other (mosaic plagiarism) or extensive fragments without indicating the source (direct plagiarism).

The 10 longest fragments (8,02 %)

Ten longest fragments found in all available resources.

NO	TITLE OR SOURCE URL (DATABASE)	AUTHOR(S)	NUMBER OF IDENTICAL WORDS	
1	Жұмабек Жанат 1.doc Satbayev University (ИКиИТ)	Жұмабек Жанат	80	1,21 %
2	Машина оқыту алгоритмдерін қолдану арқылы өмір саласы бойынша деректерді талдау Satbayev University (ИКиИТ)	Закарина Нұрзада	69	1,05 %
3	Машина оқыту алгоритмдерін қолдану арқылы өмір саласы бойынша деректерді талдау Satbayev University (ИКиИТ)	Закарина Нұрзада	64	0,97 %
4	Жұмабек Жанат 1.doc Satbayev University (ИКиИТ)	Жұмабек Жанат	63	0,96 %
5	http://www.rusnauka.com/12_KPSN_2014/Informatica/2_167_344.doc.htm		59	0,90 %
6	https://mirznaniy.com/a/113307-4/istoriya-sistem-raspoznava-niya-obrazov-4/		47	0,71 %

7	Халықтың цифрлік сауаттылығын диагностикалаудың web-қосымшасын әзірлеу D. Serikbayev East Kazakhstan State Technical University (ОПУМУП)	Әйіпов О. Қ.15-BTK-2	41	0,62 %
8	Машина оқыту алгоритмдерін қолдану арқылы өмір саласы бойынша деректерді талдау Satbayev University (ИКУИТ)	Закарина Нұрзада	37	0,56 %
9	Машина оқыту алгоритмдерін қолдану арқылы өмір саласы бойынша деректерді талдау Satbayev University (ИКУИТ)	Закарина Нұрзада	34	0,52 %
10	https://mirznaniy.com/a/113307-4/istoriya-sistem-raspoznaniya-obrazov-4/		34	0,52 %

from RefBooks database (0,00 %)

All fragments found in RefBooks database, which contains over 3 millions academic text including scientific articles published in popular databases.

NO	TITLE	AUTHOR(S)	NUMBER OF IDENTICAL WORDS (NUMBER OF FRAGMENTS)
----	-------	-----------	---

NO SIMILARITIES FOUND

from the home database (12,18 %)

All fragments found in the database of your institution.

NO	TITLE	AUTHOR(S)	INDEXING DATE	IDENTICAL WORDS (FRAGMENTS)	
1	Жұмабек Жанат 1.doc Satbayev University (ИКУИТ)	Жұмабек Жанат	2019-05-08	434 (24)	6,59 %
2	Машина оқыту алгоритмдерін қолдану арқылы өмір саласы бойынша деректерді талдау Satbayev University (ИКУИТ)	Закарина Нұрзада	2019-05-03	325 (12)	4,93 %
3	Сұлулық салонына жазылуды ұйымдастыратын "Ве.Сұлу" веб қосымшасын құру Satbayev University (ИКУИТ)	Қожантаева Бақыт	2018-05-05	43 (4)	0,65 %

from the Database Exchange Program (1,38 %)

All fragments found in the database of other institutions.

NO	TITLE DATABASE NAME	AUTHOR(S)	INDEXING DATE	NUMBER OF IDENTICAL WORDS (NUMBER OF FRAGMENTS)	
1	Халықтың цифрлік сауаттылығын диагностикалаудың web-қосымшасын әзірлеу D. Serikbayev East Kazakhstan State Technical University (ОПУМУП)	Әйіпов О. Қ.15-BTK-2	2019-05-24	41 (1)	0,62 %
2	Жеке идентификациялау негізінде қауіпсіздік жүйелерінің мәселелерін зерттеу Kostanai State University A.Baitursynov (Азғарно-технический институт)	Міркемел Б.М	2020-05-10	33 (2)	0,50 %
3	INTERWENCJE PROSUMENCKIE W INTERECIE WOBEC ORGANIZACJI – STUDIUM SOCJOLOGICZNE - część 2 SWPS Uniwersytet Humanistycznospołeczny (Szkoła Wyższa Psychologii Społecznej Uniwersytet Humanistycznospołeczny)	Magdalena Tokaj	2019-05-29	17 (1)	0,26 %

from the Internet (3,17 %)

All fragments found in the open access global internet resources.

NO	SOURCE URL	IDENTICAL WORDS (FRAGMENTS)	
1	https://mirznaniy.com/a/113307-4/istoriya-sistem-raspoznaniya-obrazov-4/	113 (3)	1,72 %
2	http://www.rusnauka.com/12_KPSN_2014/Informatica/2_167344.doc.htm	59 (1)	0,90 %
3	https://www.enbek.kz/docs/kk/node/197	17 (2)	0,26 %
4	http://kk.convdocs.org/docs/index-124103.html	13 (2)	0,20 %
5	http://www.roo-mugalzhar.kz/index.php/kz/2014-04-21-15-19-38	7 (1)	0,11 %

Протокол анализа Отчета подобия Научным руководителем

Заявляю, что я ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Елшібеков Е.Р.-Повторно

Название: 1.docxАдамдарды тану арқылы электрондық қатысу жүйесін әзірлеу

Координатор:Нұрсұлтан Таубалды

Коэффициент подобия 1:16,7

Коэффициент подобия 2:10,3

Замена букв:48

Интервалы:0

Микропробелы:0

Белые знаки: 0

После анализа Отчета подобия констатирую следующее:

- ☐ обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, признаю работу самостоятельной и допускаю ее к защите;
- ☐ обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;
- ☐ обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, не допускаю работу к защите.

Обоснование:

.....

.....
Дата

.....
Подпись Научного руководителя

АНДАТПА

Адамдарды тану бейнелерді анықтау және ақыл-ойдың жасанды шығу тегі саласындағы жетекші бағыттардың бірі болып саналады. Ол жеке куәлік, биометрия және бақылау ретінде пайдаланудың кең спектрінде қолданылған. Бұл жұмыс осындай зерттеу мен сынақ жүргізумен байланысты проблемаларды көрсетеді. Анықтау үшін бет атрибуттарының әсері, мысалы поз, шама немесе шаю зерттеледі. Салыстырмалы жоспардың салыстырмалы қорытындыларын алу үшін Мен сол әдістер тестіленетін белгілі бір деректер жинағын ойлап таптым. Нақты уақытта компьютерлік қарап шығудың барлық әдістері осы қиындықтарды шешу үшін арзан, осы жұмыс сияқты түсінікке алу үшін осы техниканың математикалық топырақтарын зерттейді, бетті анықтау үшін ақпаратты өңдейді, ал одан кейін бірнеше бет деректерінің негізі бар қосымшаның сенімділігін сынау үшін қандай зерттеулер жүргізілді. Осылайша, әдістің күшті және күшті емес жақтарын саралауға болады. Бұл жұмыста OpenCV кітапханасын енгізумен Python базасында іске асыру ұсынылады. Аппараттық қамтамасыз етудің сипаттамасы және бағдарламалық қамтамасыз етудің сипаттамасы берілген. Кейіннен еңбекшілердің кеңістіктерін дамыту жөніндегі қорытындылар, лимиттеу, жұмыстар мен шешімдер ұсынылған.

Негізгі мәтін бет анықтау, бетті анықтау, жүйе, opencv, жеке бет, нейрондық желілер, бетті анықтау әдісі, компьютерлік көру, суреттерді өңдеу, Python.

АННОТАЦИЯ

Распознавание людей считается одним из ведущих направлений в области определения образов и искусственного происхождения разума. Он был применен в широком спектре использования в качестве удостоверения личности, биометрия, и контроль. Данная работа демонстрирует проблемы, связанные с проведением такового изучения и испытания. Для определения изучается эффект атрибутов странички, к примеру поз, величина или же смыл. Дабы получить сравнительные итоги сравнительно плана, Я придумал определенный комплект данных, на которые те же методы тестируются. В реальное время почти все методы компьютерного просмотра исследуют математические почвы данной техники, дабы взять в толк, как данная работа, дешевая для заключения данной трудности, обрабатывает информацию для определения странички, а вслед за тем были проведены кое-какие исследования для испытания надежности приложения с основанием данных нескольких страничек. Этим образом, возможно квалифицировать крепкие и некрепкие стороны метода. В данной работе предлагается воплотить в жизнь на базе Python с внедрением библиотеки OpenCV. Дано описание аппаратного обеспечения и описание программного обеспечения. В последующем представлены итоги, лимитирования, работы и решения по развитию трудящихся пространств.

Главные текста определение странички, определение лица, система, opencv, собственная страничка, нейронные сети, метод определения лица, компьютерный просмотр, обработка изображений, Python.

ABSTRACT

Recognition of people is considered one of the leading trends in the field of determining the images and artificial origin of the mind. It has been applied in a wide range of uses as identity card, biometrics, and control. This work demonstrates the problems associated with conducting such research and testing. To determine the effect of page attributes, for example, POS, value, or flush. In order to get comparative results from the plan, I have come up with a specific set of data that the same methods are tested on. In real time, almost all computer viewing methods explore the mathematical foundations of this technique in order to understand how this work, which is cheap to conclude this difficulty, processes information to determine the page, and then some research was conducted to test the reliability of the application with the basis of data from several pages. In this way, it is possible to qualify the strong and weak sides of the method. In this paper, it is proposed to implement a Python-based implementation of the OpenCV library. The description of hardware and software is given. The next section presents the results, limits, work and solutions for the development of working spaces.

Main text page definition, face definition, system, opencv, custom page, neural networks, face detection method, computer viewing, image processing, Python.

МАЗМҰНЫ

КІРІСПЕ	10
1 Бейнелерді тану міндеттерінің сипаттамасы және түрлері	11
1.1 Жалпы сипаттама	11
1.2 Бағдарламалық модельдерді оқыту алгоритмдерінің сипаттамасы және түрлері	12
1.3 Бет әлпетті айырып тану әдістері	12
1.4 Бейнелерді таңдаудың қазіргі жағдайы мен перспективалары	14
1.5 Бейнелерді танудың қазіргі жағдайы мен перспективалары	14
2 Бейнелерді тану бағдарламасын құрудың компоненттері	16
2.1 OpenCV кітапханасын қолдану тиімділігі	16
2.2 Microsoft visual studio интегралданған бағдарламалау ортасын қолдану	17
2.3 Python бағдарламалау тілі	18
2.4 Python үшін Django фреймворк	20
3 Бет әлпетті тану алгоритмі	23
3.1 Бетті танудың басты міндеттері	23
3.2 Бет кеңістігінің базисін құру	23
3.3 Абстрактілі қойылымы	24
4 Бағдарлама жасау үшін пайдаланылатын құралдар	26
4.1 Басты құралдардың негізгі сипаттамасы	26
4.2 MVC архитектурасы	28
4.3 Қолданбалы бағдарламалық қамтамасыз ету	30
4.3.1 Google Chrome Браузері	30
4.4 Аспаптық бағдарламалық қамтамасыз ету	31
4.4.1 PyCharm әзірлеу ортасы	31
4.5 Бағдарламалық қамтаманың көрсетілімі	32
ҚОРЫТЫНДЫ	36
А қосымшасы	38
Б қосымшасы	44
С қосымшасы	48

КІРІСПЕ

Берілген дипломдық жұмыста адамдардың бейнелерін анықтау және идентификациялау әдісі ұсынылады. Анықтау әдісі тек үш өлшемді бейнелерді ғана пайдаланады және түсі туралы ақпаратты және тек бет стилі туралы ақпаратты пайдаланбайды. Мимикалық экспрессиялардың және субъектілердің басты жағдайының өзгеруін қамтитын адамдардың үш өлшемді беттерінің нақты деректер базасын енгізумен адамдарды анықтау қорытындылары келтіріледі.

Мен жасаған бейнелерді анықтау жүйесі үшін бағдарламалық қамтамасыз етудің басты мақсаты емтихан тапсырар алдында оқушылар үшін бақылауды қамтамасыз ету, сабаққа қатысу, сабаққа қатысу, ұсынылған бейнелердің қандай даралығын енгізе отырып, сабақтарға қатысу болып саналады. Сонымен қатар, бұл жүйенің сенімділігін және оларға әсер ететін сәттерді, сондай-ақ қойылған сәттердің әсерін төмендету қабілетін зерттеу, бейне ағыннан адамдарды анықтау бағдарламасын жазу деп саналады.

Тапсырманың салмақты әдістемелерінің бірі жиынтық функцияларды енгізу болып саналады. Біз мойындайтын үлгілер туралы кішкентай априорлы деректер болған кезде, анықтау жүйесін әзірлеу кезінде зерттеу функциясын пайдалану қадамдары бойынша жүреді. Бірінші шепте ерікті бөлу функциялары таңдалады және бөлу функциясын зерттеудің итеративтік қадамдарын орындау барысында қолданылатын келбетке дейін орын алады.

Адамдарды білу-адам үшін өте қарапайым мәселе. Тәжірибе барысында 1-3 күн балалары өздерінің адамдарын ажыратуға барлық мүмкіндіктері бар екенін көрсетті. Бірақ бұл компа үшін қарапайым емес. Сыртқы (шаш, бас конфигурациясы) немесе ішкі (көз, мұрын, ауыз) белгілері қандай? Суретті қалай оқып, оны кодтау керек? Ғылыми қызметкерлердің Думасына 1-ші орналасу жазықтықтың геометриялық суреттерінде жүзеге асырылған. Және 1 белгіленген нүктелер (ұшы, құлақ, мұрын жағдайы) символдар векторын құру үшін қолданылған (нүктелер арасындағы қашықтық, олардың арасындағы бұрыш) осы бастапқы өзін-өзі әсер ететін адамдарды бөлу жүйелерінің ішінен. Ев-клидтердің кіріс және тірек бейнесінің сигнал векторлары арасындағы қашықтықты табу үшін білімді өзі орындады. Бұл әдіс Жарық конфигурациясына төзімді, бірақ белгіленген нүктелерді дәл анықтау мүмкіндігі жоқ. Бұл тәсілдің жетекші тәсілі жетекші компонент (PCA - Principle component analysis) тәсілін қолдана отырып, ақпараттық мазмұнның маңызды шығынынсыз бастапқы бейненің инфасын қысу болып табылады. Көпмәдениетті болдырмау үшін символдар тек жақын жерде ғана суреттей бастады-неге, бұл символдар кедергілерге, жарыққа және іріктеудің ең аз көлеміне көбірек төзімді болуға міндетті. Бұл әдістерге мыналар жатады: габаритті вейвлеттер, дискретті косинусты қайта орнату, аудандық бинарлы паттерналар.

Адамдардың бейнесін анықтау базасындағы рәсімдерге назар аудару, әсіресе өсіп келе жатқан практикалық қажеттілікке байланысты: қорғау жүйелерін сынау, сот сараптамасы, телеконференция және т. б.

1 Бейнелерді тану міндеттерінің сипаттамасы және түрлері

1.1 Жалпы сипаттама

Сипаттамаларды анықтау-кластарды (бейнелерді) анықтау немесе анықтау үшін жүйені әзірлеудің концепциясы мен негіздерін зерттейтін, зияткерлік белгісіз объектімен, сигналмен, түрлі іс-шаралармен, пайда болуымен немесе процестермен информатиканы бөлу. Сыныптамалық міндеттерді жасасу үшін автомобильдерді қолдану мүмкіндіктерін тергеуде таңу аяқталады. Беневиітерді тағайындау қорғаныс өлшемдерінде ретсіз қолданылады. Бейнелерді таңдау көлемі жалпы жетекші (белгілері, қасиеттері) сипатталады. Негізгі мүмкіндіктер әр түрлі ерекшеліктерді меңгеруге барлық мүмкіндіктерге ие болады: олар табиғи тіктелген жинақтан немесе дискретті жинақтан болуға барлық мүмкіндіктерге ие. Бұл объектіні түсіну практикалық қосымшалардың қажеттілігі ретінде, мысалы, сипаттамаларды анықтау, мысалы, объектіні қабылдау механизмімен біздің ұғынуымызбен келісіледі. Компьютер көп мөлшерде қайталанбайтын біртектес құрамдастардан тұрады. "Зақымдалмаған" заттың болуын немесе қол жетімсіздігін білдіреді.

Адамның жеке деректерін пайдалану ойы жаңалық емес. Қазіргі уақытта қорғау жүйелерінде технологиялар бар, қихт-да тұлғаны сәйкестендірудің барлық мүмкіндіктері қолданылады. Олар:

- саусақ ізі (жеке және абсолютті жазылу);
- сыртқы бет бейнесі (оптикалық және инфрақызыл бейнелер негізінде);
- марокко тану;
- штрих-кодтарды анықтау;
- авто нөмірлерді анықтау;
- сөзді анықтау;
- алаяқтықты анықтау;
- мәтінді анықтау;
- спамды анықтау;
- құжаттарды санаттау;
- қолжазбаларды анықтау;
- сипаттамаларды анықтау;
- жер қыртысының аудандық учаскелерін тану;
- басқа да қасиеттер бар.

Барлық биометриялық технологияларды сәйкестендіру мәселелерін шешудің өзіндік мүмкіндіктері мен әдістері бар. Дегенмен, барлық мүмкіндіктер досынан қолдану тиімділігі мен қорытынды дәлдігіне бөлінген. Әрбір биометриялық өңдеу кезең-кезеңмен қолданылады:

- объектіні сканерлеу;
- өз ақпаратын алу;
- үлгілер жинағы;
- ағымдағы үлгіні деректер базасымен салыстыру.

Биометриялық анықтау жүйесі нақты үлгінің сәйкестігін анықтайды .

Биометриялық жүйелерді, оның ішінде жазықтығы бойынша пайдалану кезінде және дұрыс биометриялық деректер туындаған жағдайда сәйкестендіру туралы қорытынды барлық рет оң болып саналмайды. Бұл жалпы жағдай, бастау үшін, биометриялық деректерді ауыспалы көбейтумен тығыз байланысты. Жүйе промахи ықтималдығының бәсекелестік дәрежесін қамтиды. Данияда, әр түрлі технологияларды пайдалану кезінде жарамдылықтың бөліну мүмкіндігі бар. Енуді бақылау жүйесі үшін, биометриялық технологияларды пайдалану кезінде алдымен дәрежеліу немесе елеулі "кедергіні" өткізбеу немесе барлық"жеке адамды"жүргізу қажет.Қорғау жүйелеріндегі биометриялық технологиялардың юзерлері үшін қажетті сәт пайдаланудың қарапайымдылығы болып саналады. Өзінің сканерленетін тұлғалары ешқандай ыңғайсыздық сезінуге міндетті емес. Бұл жағдайда, мүлдем, адам бойынша анықтама әзірлеу неғұрлым қызықты әдіс болып саналады.Адамдардың бұрмалануын тану қорғауды қамтамасыз етудің аса жоғары талаптары бар объектілерге, мысалы, банктерде немесе схемалық кәсіпорындарда қолданылады. Модульді басқа да қажетті пайдалану-Казинода, қонақ үйлерде, мейрамханаларда және басқа да осыған ұқсас объектілерде фейс-бақылауды автоматтандыру. Бірнеше фирмалар 40 жылдан астам уақыт бойы Автоматты, қазіргі заманғы адамдарды бөлу жүйесін тиімді әзірлейді: Smith & Wesson (ASID-Automated Suspect Identification System); Imageware (fase жүйесі); Imadis, Eric Solutions, Spillman, Miros (Truefase жүйесі); Vissage Teshnology (vissage Gallery жүйесі). жүйесі); visionisss (базалық жүйе).

1.2 Бағдарламалық модельдерді оқыту алгоритмдерінің сипаттамасы және түрлері

Тривиальды көрнекті қасиеттерге назар аудармай, биометриялық деректер сұрақтар туғызатын биометрияға қарсы дұрыс емес пікірлердің жағымсыз тізбегіне ие, биометриялық деректер адамның өзіне және өз өміріне қатысты олардың құқықтарын сақтамауға барлық мүмкіндікке ие. Биометриялық технологиялар істің нақты оқиғаларынан ерекшеленіп отырады. Бұл ең аз емес, соңғы жылдары биометриялық сәйкестендіру тәсілдерін пайдалану ерекше өзектілікке ие болды. Оның үстіне, АҚШ-та 11 қыркүйектегі іс-шаралар нәтижесінде осы мәселеге қамқорлық берілді. Бүкіл әлемдік бірлестіктер бүкіл әлемде терроризм қаупінің ұлғаю дәрежесін және классикалық тәсілдермен сенімді қорғанысты ұйымдастырудың күрделілігін қабылдады. Бұл қайғылы әрекеттер прогрессивті кіріктірме қорғаныс жүйелеріне деген қызығушылықты арттыру үшін алғышартқа айналды. Егер әуежайлардағы бақылау күшейтілсе, онда көптеген қолайсыз жағдайларды болдырмау мүмкін болар еді. Нақты уақытта оқиғаға кінәлі адамдарды барлау бетті анықтау жүйелерімен орнатылған алдыңғы қатарлы Бейнебақылау жүйелерін енгізумен жеңілдетуге мүмкіндік береді.

1.3 Бет әлпетті айырып тану әдістері

Нақты уақытта бет анықтаудың 4 жетекші бейнесі бар:

- eigenfases;
- ерекше белгілерді саралау;
- нейрондық желі негізінде дифференциация;
- бет-бет бейнесін өз бетінше өңдеу тәсілі іске асыру күрделілігімен және жоғарыда келтірілген барлық тәсілдерді пайдалану мақсаттарымен ерекшеленеді.

"Eigenface" мәтінін "өзіндік тұлға" ретінде жылжытуға болады. Бұл жоба бет-фигураның бейнесінің өзіндік қасиеттерін білдіретін сұр градацияның екі өлшемді бейнелерін пайдаланады. "Eigenface" әдісі басқа тұлғаларды анықтау үшін топырақ ретінде қолданылады.

"Eigenface" 100 - 120 қасиеттерін біріктіре отырып, адамдардың үлкен санын қалпына келтіруге мүмкіндігі бар. Тіркеу кезінде кез келген белгілі бір адам "eigenface" коэффициенттерінің тізбегі ретінде береді. Сурет арақатынасты сынау үшін қолданылатын түпнұсқалықты анықтау тәртібінде қалғандарының коэффициентін анықтау мақсатында бекітілген "нақты" прототипімен салыстырылады. Эталондар арасындағы Өзін-өзі тану дәрежесін теңдеу коэффициентін анықтайды. Бетті сканерлеу қабілеті болған жағдайда тамаша жарықтандырылған үй-жайларда "eigenface" технологиясын дұрыс қолдану. "Ерекше симптомдарды" саралау әдісі-бұл кең қолданылатын идентификацияны дамыту. Бұл әзірленім "eigenface" тәсіліне ұқсас, бірақ жеке тұлғаның немесе сыртқы келбетінің (күледі немесе әлсіреген тұлға) өзгеруіне бейімделген. "Ерекше белдіктер" технологиясында жазықтықтың әр түрлі аймақтарындағы 10 өзіндік даралық қолданылады, сонымен қатар шартты орналасуын ескере отырып, осы сипаттамалардың дербес логикасы кез келген белгілі бір тұлғаның ерекшелігін анықтайды. Жеке тұлға қайталанбас, бірақ өзгереді. Адамның ұйықтауға мүмкіндігі бар болғандықтан, көзілдіріктің барлық мүмкіндігі бар, мұртты және сақалды жіберу-осының бәрі сәйкестендіру режимінің қиындықтарын арттырады. Осылайша, мысалы, араластыру кезінде бет бөлігінің ығысуы бар. Осы жылжуларды ескере отырып, жеке тұлғаның әртүрлі конфигурацияларында сәйкестендірілуі мүмкін. Қазақстан Республикасының 2007 жылғы 27 шілдедегі № 319 - III "білім туралы" Заңының 9 - бабының 1 - тармағына сәйкес, Қазақстан Республикасы Президентінің 2010 жылғы 29 қарашадағы № 1113 Жарлығымен бекітілген Қазақстан Республикасында білім беруді дамытудың 2011-2020 жылдарға арналған мемлекеттік бағдарламасын іске асыру мақсатында, Қазақстан Республикасы Президентінің Жарлығымен бекітілген Қазақстан Республикасында білім беруді дамытудың 2011-2020 жылдарға арналған мемлекеттік бағдарламасын нейрон желісіне негізделген тәсілде іске асыру мақсатында тіркелген және қорғалатын "Нейрондық желілер" деректер негізіндегі модель сипаттамаларының және тексерілетін тұлғаның эксклюзивті параметрінің арақатынасын нақтылайтын әдісті пайдаланады, сондай-ақ сипаттамалардың ықтимал саны қолданылады. Сәйкестендіруге қарай деректер негізінде прототип арасындағы және тексерілетін тұлғаның сәйкессіздігі анықталады, кейіннен тиісті салмақ коэффициенттерін қолдаумен тексерілетін тұлғаның деректер негізінде эталонға арақатынасының дәрежесін айқындайтын құрылғы орындалады. Бұл

әдіс қиын жағдайларда тұлғаны сәйкестендіру сапасын арттырады.

1.4 Бейнелерді тандаудың қазіргі жағдайы мен перспективалры

"Бет бейнесінің өзі әсер ететін өңдеу" әдісі-бет нүктелері мен жазықтықтарының ара қашықтығы мен қашықтығын, мұрын ұштары, ауыз бұрыштары ретінде анықталатын қарапайым әзірлеу.

Алайда, бұл әдіс "eigenfaces" немесе "нейрондық желі" сияқты қуатты емес, шын мәнінде өте хилом жарықтандыруда қолданылуы мүмкін. Бұл әдіс көптеген жағдайларда, мысалы, қойылған қағаздарды ұсынған адамға бейімдеуді анықтау кезінде қолданылады. Жауап беруші: жургараев Ермек Нурланович бұл қателік егер де қазіргі пайдаланушы "қауіпсіздік" атты қосымша бетінде ақпараттық база бумасында қойылмаған болса немесе бумаға ортақ қатынаудағы пайдаланушылар тізімінде болмаса шығады. Бұл әдіс бойынша компьютерлік қатынау жүйесі жұмыс істейді. 2-і айырмашылығы, компьютерлік қол жетімділік жүйесінде фото деректер негізінде сақталатын прототиппен салыстырылады.

Мұндай жағдайларды санамағанда, инфрақызыл жарықтандыруда бетті анықтау технологиялары байқалды. Жаңа зерттеме беттің қан тамырларының жылуымен бет жазықтығының термограммасы немесе басқаша айтқанда, термиялық сурет жасалған бет жазықтығының термограммасы болып табылады. Бұл кескін жүйенің кез келген юзері үшін жарамсыз болып саналады және енуді бақылау жүйелері үшін биометриялық деректер ретінде қолданылуы мүмкін. Берілген термограмма жазықтықтың геометриясына қарағанда өзгеріссіз жеке нөмір болып саналады. Себебі, бұл тұлғаның өзгеруімен байланысты емес.

1.5 Бейнелерді танудың қазіргі жағдайы мен перспективалары

Адамдардың білім беру тапсырмасын жасау кезінде 2 сұрақ пайда болады. Алдымен, кез келген нобай пиксель массиві болып саналады. Мұндай пиксель кескін санамайды, ештеңе білдірмейді (оның бояуы өзгеруі мүмкін және ешкім айырмашылықты таба алмайды). Бұл көріністі анық және қаржылық емес дайындайды. Осылайша, адамдарды тиімді анықтау үшін кескіндерді ұсынудың шағын және қолайлы пішімін жасау керек. Қазіргі денекке жоғалған суреттерді қысу әдістемелерінің көп саны хабарланған, бірақ онда қолданылатын формат адамдардың бейнелерін жүйелендіру үшін қолайлы емес, мысалы, адамдарды анықтау қиындықтарын шешу үшін тағы да ақпарат қажет.

Бұл бірінші кезекте осы адам коллекциядан қалай көрінеді деп саралаудың қажеті жоқ, ал айналым тапсырмасын шешу керек: коллекциядан қандай адам осы жолмен көрінеді.

2-ші мәселе-бір сурет жарықпен байланысты, әсерге байланысты, сенімге байланысты барлық мүмкін себептер бойынша бекітілуі мүмкін.

Шын мәнінде, адам объектіні БАҚЫЛАУДА ұстағанда (өлшеуде), ол туралы деректер сенсорлардың соңғы саны бойынша миға түседі (талданатын арналар) және кез келген детекторға объектінің тиісті қасиеттерін салыстыруға

болады. Біздің объектіміздің тиісті аспектілеріне символдарды санамағанда, бөлінген символдар немесе жіктеу символдары деп аталатын символдар тобы да бар және X векторындағы олардың мәндерін анықтау кезінде тану жүйесінің табиғи және жасанды пайда болуын жасайтын тікелей міндеттен жасалған. Бейнені анықтау барысында белгілерді бөлу және топтастыру, яғни жазбаша сөздерді тану кезінде пайда болатын ең кең міндеттердің бірі - 1-нің жеке белгілеріне графикалық нысандағы сөзді фрагменттеу.

2 Бейнелерді тану бағдарламасын құрудың компоненттері

2.1 OpenCV кітапханасын қолдану тиімділігі

OpenCV (Британдық тілдің бастапқы көзі жабық емес компьютерлік жағымды кітап сақтау орны, бастапқы коды жабық кітап сақтау орны) - компа көру, суретті өңдеу және бастапқы кодты ашу үшін ортақ пайдаланылатын сандық әдістер. C / C ++Python, Java, Ruby, Matlab, Lua және басқа тілдер үшін жасалған. Оқу және ақылы мақсаттар үшін еркін қолдануға болады-BSD лицензиясының өлшемдеріне қолданылады.

Осы саладағы қосымшалар үшін жалғыз қалыпты компа көру пайдаланушы интерфейсін нығайту үшін. Бұл осы қосымшалардың санын өсіруге және компа қолданудың жаңа үлгілерін жасауға рұқсат етеді.

Intel Intelligence Libraries және MKL (LAPACK және FFTPack арнайы нұсқасы) қолдайтын OpenCV жылдамдығына байланысты осы бағдарламаларды жасаушылар үшін Intel платформасын әдемі етіп орнатыңыз. OpenCV IPP және MKL болуын механикалық алдын ала анықтауға және оларды өңдеуді жылдамдату үшін қолдануға мүмкіндігі бар.

Қолдау көрсетілетін префикстер мен құралдар кітапхананың өздері:

- microsoft windows: Microsoft visual c ++ (6.0, .net 2003), intel compiler, borland c ++, mingw (gcc 3.x);
- windows rt: iteez арқылы arm [3];
- linux: gcc (2.9x, 3.x);
- mac os x: gcc (3.x, 4.x);
- android;
- ios;
- c және жеңіл салмақты c қолданылады. прагма және салыстырмалы компиляция өте c;

GUI құралдары, бейне түсіру:

- Linux: V4L2, DC1394, FFMPEG;
- Mac OS X: QuickTime.

Құжаттау: статикалық HTML, PDF. Негізгі Модульдер.

2.2 нұсқасында кітап қоймасы қайта құрылды. Sxcore, cvaux, highGUI және басқа әмбебап модульдердің орнына, тар шағын габаритті модульдердің кейбір Саны жасалды: негізгі жұмыс қабілеттілігі opencv_core. Негізгі құрылымдар, есептеулер (математикалық функциялар, кездейсоқ сандар генераторлары) және сызықты алгебра, DFT, DCT, XML / YAML және т. б. үшін Енгізу / шығару :

- opencv_imgproc-суретті өңдеу (сүзгілеу, геометриялық конфигурациялар,

түс орнын қайта орнату және т.б.);

- opencv_highgui - қарапайым пайдаланушы интерфейсі, суреттер мен бейнелерді енгізу / шығару;

- opencv_ml-машиналарды зерттеу модельдері (SVM, қорытынды ағашы, стимуляция және т.б.);

- opencv_features2d-жазық примитивтерді тану және сипаттау (SURF (Eng.) Ресей, кесу және өзге де арнайы негіздер саны);

- opencv_video-объектілерді жылжыту және бақылау (оптикалық ағын, жылжу үлгілері, фонды жою);

- opencv_objdetect-суретте объектілерді анықтау (Виола-Джонс әдісін енгізумен Хогта адамдарды тану және т.б.);

- opencv_calib3d-бейнекамераны калибрлеу, стереополимерлерді іздеу және үш өлшемді деректерді өңдеу элементтері;

- opencv_flann-жақын көршілерді жылдам барлау (FLANN 1.5) және OpenCV орау;

- opencv_contrib-қолдануға дайын емес дос коды. opencv_legacy-ескірген сәйкестікке сақталған ескірген код;

- opencv_gpu-Nvidia көмегімен жасалған CUDA рахмет OpenCV қандай функцияларды жылдамдату.

2.2 Microsoft visual studio интегралданған бағдарламалау ортасын қолдану

Visual Studio-Microsoft ұсынған әртүрлі экстрa класты буындары үшін әзірлеудің интеграцияланған жағдайы.

Visual Studio бағдарламашыларға Win32 / Win64 бағдарламаларын ұйымдастыруға және қосуға рұқсат береді . NET Framework. Сонымен қатар, Visual Studio Windows бағдарламалары динамикалық веб-сайттар мен интернет / Интранет немесе Azure қызметтеріне арналған веб-қызметтер көрсетіледі. Қосымша фокус-покус Windows Phone, Android және Xamarin құралдарын қолдайтын учаскеге арналған жылжымалы қосылымдарды дамытуда қамтылған.

Xamarin негізіндегі Mac үшін Visual Studio, бұл MacOS үшін Visual Studio нұсқасының бірінші нұсқасы болуы мүмкін. NET 10 мамырда Xamarin Studio құрылған, 2017 [4], Басты және бірлік, ойындар мен қосымшаларды қалыптастыру. Мақсатты платформалар-Android, macOS, iOS, Web және Cloud. Фокус-покус C# және F#, және HTML, CSS және JSON.

Visual Studio көптеген мүмкіндіктер бар қолайлы даму саласы болып табылады. Мысалы, курсордың ватерпас үшін УДЗ орнату код агрегаттарын тереңдету мен жоюды, кілттердің түрлі-түсті таралуын, өзіндік әсер ететін синтаксис апробациясын және IntelliSense өлеңін енгізу кезінде функцияны жасауды, бейсаналық рецепт және кодты енгізуді қамтиды. Сонымен қатар,

өңдеу жағдайы ActiveX және кітапханаларын интеграциялау үшін графиктік интерфейстер, деректер бастау үшін "сервер зерттеушісі" болып табылады. NET, веб-қызметтер: мысалы, Microsoft SQL принтсервері, Windows жаңалықтарының жұмысы WMI сиректеу, және Windows пайдаланушы интерфейсін және Веб-қосымшаларды жасау үшін XML белгішелері мен құжаттарын өңдеу үшін WYSIWYG редакторлары.

Visual Studio Бағдарламасы. NET (2002), қолдау көрсетілетін буындардың спектрі Мұқият кеңейтілді. Microsoft ұйымының аттестаттау коды бөгде жеткізушілерге Visual Studio бағдарламасында қосылған яко буындарын ұсынуға рұқсат береді. Образчики Делфи, Eiffel, F# сиречь Proolog. Visual Studio 2008 нұсқаларын Microsoft Visual J#қолдауы жоқ.

Visual Studio біріктірілген түзету құралын қамтиды. Бұл "редакциялау және жалғастыру" функциясын қамтиды және жергілікті компьютерде және желіде одан әрі іске қосылған процестер үшін беріктілік қорын ұсынады.

ASP. NET Visual Studio Version 2005 немесе одан да көп кешіккен нұсқалар интеграцияланған веб-серверді қамтиды, одан бөлек IIS станциясы сұралмайды. Иә Refactoring, қосымша провианттармен ("қондырмалар"), соның ішінде Reactor, ReSharper refactor осы нұсқа үшін күреседі! Pro қайта құру болды.

Visual Studio қолданбаларды әзірлеу үшін көптеген редакторларды шақырады. Қандай маңызды:

- Жоғарыда аталған бағдарламалау буындары үшін кодтар редакторлары.
- Графикалық интерфейс редакторлары.
- Ақпараттық базаның схемалары / деректер негіздері.
- HTML веб-беттеріне арналған редакторлар.
- XML, XSLT, слог кестелері және XML негізіндегі әртүрлі конфигурация файлдары үшін редакторлар.
- XSD БИЛЬДРЕДАКТОРЫ.
- Графикалық редакторлар.
- Екілік Файл редакторлары.
- Құрылғыларды пайдалануды орнатушыларға арналған редакторлар (графикалық).
- Сынып диаграммасының бильдредаторы.
- Схема бағдарламасы факультативтік редактор.

Барлық мәтіндік редакторлар синтаксистің апробациясын (бастапқы кодтың дұрыстығы), синтаксистің таратылуын, IntelliSense (элементтердің Автоматты нұсқауын) және автотолдауды (элементтердің автоматты түрде бөгеуін) сақтайды.

2.3 Python бағдарламалау тілі

Python-ең танымал қазіргі бағдарламалау буындарының бірі. Ол әр түрлі мәселелерді шешу үшін қолайлы және осы функцияларды ұсынады, оның ішінде басқа бағдарламалау слогдары: динамик, ООП көмек және кросс-

платформалық. Python әзірлемесін 1990-ші жылдардың жартысында Гвидо Ван Россум (Guido Van Rossum) белсендірді, өйткені қазіргі уақытта қарапайым "балалар" аурулары, үздік слог жақтауларының әсерлі қалыптасуы және Python өз жобаларын іске асыру үшін кәдеге жарататын көптеген бағдарламашыларды қатыру.

Мысалы, Python бағдарламалық қамтамасыз ету алаңында өз аяқталуын бастамайды, ал Python өте жақсы" кіріспе " бағдарламалау тіліне айналады. Өзінің қысқа мерзімділігінің салдарынан ол тілдің синтаксисін проворно жерсіндіруге мүмкіндік береді, ал" мұраның " болмауы көптеген ұшулар кезінде қалыптасқан аксиом ООП-ны проворно жерсіндіруге көмектеседі. Осы факторларға қатысты, Python "оқыту қисығы" шексіз қысқа болады және программист оқу модификацияларынан бизнесмендік жобаларға қылмыс жасауы мүмкін.

Python Құрылымы. Әр тілдің маңызы жоқ-кем дегенде екі бөліктен тұрады бағдарламалау үшін қарым-қатынас-сөздіктер және синтаксис. Python диалектісі ойнатылатын бағдарламаларды құрайтын өрнектерді қалыптастыру үшін синтаксисті жеткізеді және ұйымдастырылған, сөздік-қарапайым кітапхана мен қосылатын модульдер нұсқасындағы функционалдылық жиынтығы.

Жоғарыда айтылғандай, Python синтаксисі шексіз қысқа, әсіресе Java немесе с++салыстыру болса. Бір жағынан-бұл жақсы, мысалы, синтаксисдің себептері оны пайдалану барысында жеңілірек, соғұрлым аз қателіктерді зерттеу жақсы. Дегенмен, мұндай буындар жеткіліксіз көрінеді – олар ең қарапайым ақпаратты күрделі құрылымдарды көрсетпеуге мүмкіндік бере алды. Python бұған есептелмейді, бұл диалект қарапайым, бірақ жеңілдетілген. Python, мысалы, жоғары, Java және С++ сияқты абстракцияның асыл ватерпасы бар тіл болып табылады және мұндай ақпаратты бастапқы кодтың шағын мөлшерінде таратуға мүмкіндік береді.

Иә Python корпоративтік мақсаттағы тіл ретінде ұсынылады (standalone, клиент-сервер, Web-қосымшалар) және кез келген осы салада. Сонымен қатар, Python қолда бар компоненттермен еркін интеграцияланады, бұл Python-ды Ұлттық бағдарламаға қосуға мүмкіндік береді.

Python табыстың қалған ингредиенттері-қарапайым және арнайы кеңейтулердің модульдері. Python кеңейтуінің қарапайым модульдері-БҚ әзірлеу бойынша кез келген жоспарда байланатын міндеттерді шешу үшін тамаша жобаланған және бірнеше рет сыналған жұмыс қабілеттілігі, жолдар мен мәтіндерді өңдеу, операторлық жүйемен келісу, Web-қосымшалардың көмегі. Бұл модульдер Python тілінде Ұлттық нақышталған, өйткені оның маңызды қасиеттеріне ие-бір операторлық икемділік арқылы екінші операторға ауырмайтын кроссплатфор.

Егер Python қарапайым кітапханасында жеткілікті функционалдылық болмаса, онда оны одан әрі бірнеше рет қолдану үшін өзінің әріптік кеңейту торабын ұйымдастыруға болады. Python үшін кеңейту модульдері Python тілінде емес, сондай-ақ басқа бағдарламалау буындары арқылы ұйымдастырылуы мүмкін. Бұл жағдайда ресурсты қажет ететін міндеттерді

жемісті жүзеге асыру мүмкіндігі туындайды, мысалы, күрделі академиялық есептеулер, алайда, кеңею Модулінің диалектісі Python сияқты кросс-платформалық артықшылықпен жіктелмейді, кросс-платформалық артықшылығы жоғалады.

Белгілі болғандай, кросс-платформалық бағдарламалаудың барлық тілдері бір үлгіден құрылған: бұл шын мәнінде тұрақсыз алғашқы Код және жүзеге асыру жағдайы (runtime environment), кейбір белгілі бір платформа үшін ерекшеленбейді. Бұл салаға әртүрлі утилиттер – реттеуіштер, қарама-қарсы микроассемблер және т. б. орналастырылады.

Java орындау ортасына факультативті компилятор орналастырылады, алғашқы қауымдық бағдарламалардың себептілігі автор шартты Жавамашина үшін байт-кодқа орнатуы тиіс. Python орындау ортасында тек интерпретатор орналастырылады, кейбір кенеттен компилятор болып табылады, бірақ Python бағдарламасын табиғи машина кодына мақсатты платформа жасайды. Қазіргі уақытта Python үшін үш белгілі орындау саласы бар: CPython, Jython және Python.NET атаудан табу қиын емес сияқты, бірінші жағдай C тілінде, екіншісі-Java слогында, соңғысы-платформада сатылды . NET.

CPython жүзеге асыру жағдайы, көп жағдайда, Python, және кейде Денсаулық Python туралы келіседі, бірінші кезекте бұл іске асыру. Бұл іске асыру с слогына арналған интерпретатор мен кеңейту модульдерінен құралады және қарапайым компиляторы бар кез келген платформада қолданылуы мүмкін. әр түрлі операторлық жүйелер үшін жүзеге асыру саласының компилденген нұсқалары, Windows-тың барлық нұсқалары мен Linux-тың әртүрлі дистрибутивтерінен бастап. Осы және одан кейінгі жазбаларда CPython қарастырылады, егер басқаша бөлек ескерілмеген.

Jython орындау жағдайы-Java (JVM) виртуалды автомашинасымен жұмыс істеу үшін Python жүзеге асыру. Әрбір JVM опциялары 1 нұсқасынан белсендіріледі. 2. 2 (Java-1 ағымдағы нұсқасы. 6). Jython-мен жұмыс істеу үшін қойылған Java-машиналарын (Java орындау ортасын) және диалект бағдарламасын білу қажет.

2.4 Python үшін Django фреймворк

Django-бұл Python бағдарламалау слогына арналған күрделі сайттар мен веб-қосымшаларды әзірлеу үшін ыңғайлы мүмкіндіктері бар макро бағдарламалық қорап.

Django Apache (mod_python) басқарумен жұмыс істеу үшін және PostgreSQL берілген негіз ретінде жобаланған. Қазіргі уақытта, PostgreSQL басқалармен әрекет етуі мүмкін, Django SQLite (MariaDB), SQLite, Microsoft SQL Server, DB2, Firebird, SQL Anywhere және Oracle. Django берілген базалармен жұмыс істеу үшін дербес ORM кәдеге жаратады, берілген модификация Python кластарымен сипатталады және ол бойынша деректер негізі бағдарламасы ұйымдастырылады.

Django құрылымы "модель-сыртқы көрініс-Контроллер" (MVC) сияқты.

MVC антикалық модификациясының реттеуіші Django-да кескін ватерпазына (View) сәйкес келеді, ал демонстрация логикасы Django-да стандарттар (Templates) деңгейінде орындалады. Django деңгейлік архитектурасын назарға ала отырып, тәулік сайын "үлгі-үлгі-көрініс" (MTV).

Django-ның алдын ала әзірлеуі жаңалықтар ресурстарымен көбірек қолайлы қызметті қамтамасыз ету үшін жүргізілді, бұл сәулет үшін қатты әсер етті: фреймворк ақпараттық сипаттағы веб-сайттардың арандатушылық жаттығуына көмектесетін бірқатар құралдарды шақырады. Мысалы, құрастырушы сайттың басқару үлесі үшін контроллерлер мен беттерді жасауды сұрамайды, Django-да мазмұнды басқару үшін интеграцияланған кодты тұтынады, кейбір Django-да жасалған кез келген сайтқа қосуға болады және бір серверде бірнеше сайттарға бірден қосуға болады. Кеңсе юзерлер мен топтарды басқару үшін сокет береді (әрбір объект бойынша құқықтармен), сайт толтырудың әр статийін жасауға, дұрыс ойлауға және алаңдатуға, әрқашан дайындалған іс-әрекеттерді хаттамалауға рұқсат береді. Веб фреймворк Django Instagram, Disqus, Mozilla, The Washington Times, Pinterest, соның ішінде ірі және танымал сайттарда қолданылады,

Ламода және т. б.

Django қабілеті қандай:

- API өту ДҚ-да Orm, транзакциялар.
- Әкімші сокеті, бар тілдер үшін көптеген аудармалар бар.
- Ұзақ мерзімді өрнектер базасы үшін URL жетекшісі.
- Тег және тұқым қуалайтын модельдердің жетілдірілген құрылымы.
- Кешігу құрылымы.
- Интернационалдандыру.
- Кез келген Django-сайттарында анықталған қосымша сәулет.
- "Generic views" - контроллерлер функцияларын түрлендіру.
- Авторландыру және аутентификация модульдерін қосу, сыртқы аутентификация: LDAP, OpenID және т. б.

- Мысалы, қосымша сұраныстар конструкторларын құру үшін сүзгілердің конструкциясы ("middleware"), соның ішінде дистрибутивке орналастырылған сүзгілер, кешігу, сығылу, URL-ды жаңарту және жасырын сессияларды қолдау үшін сүзгілер.

- Формамен жұмыс істеуге арналған кітапхана (мұралау, қолда бар ДБ модификациясы бойынша конфигурациялар құру).

- Басқару Қосымшаларының ақылға қонымды құралымен, стандарттар мен деректер үлгілерінен кейін жіберілген өздігінен әрекет ететін құжаттама.

Фреймворк легоньконың қандай ингредиенттері жеңіске біріктірілді, өйткені олар тек ұқсас болуы мүмкін. Бірақ кейбір (мысалы, ORM) бұл шексіз оңай емес. Фреймворк ядросында аннексияланған мүмкіндіктерден басқа, оның

мүмкіндіктерін кеңейтетін пакеттер бар.

Django базасында тәуелсіз лицензиялар тарататын көптеген дайын шешімдер құрылды, интернет-дүкендерді басқарудың икемділігін, контентті басқарудың барлық тиімді икемділігін және тығыз бағытталған жобаларды жеңемін.

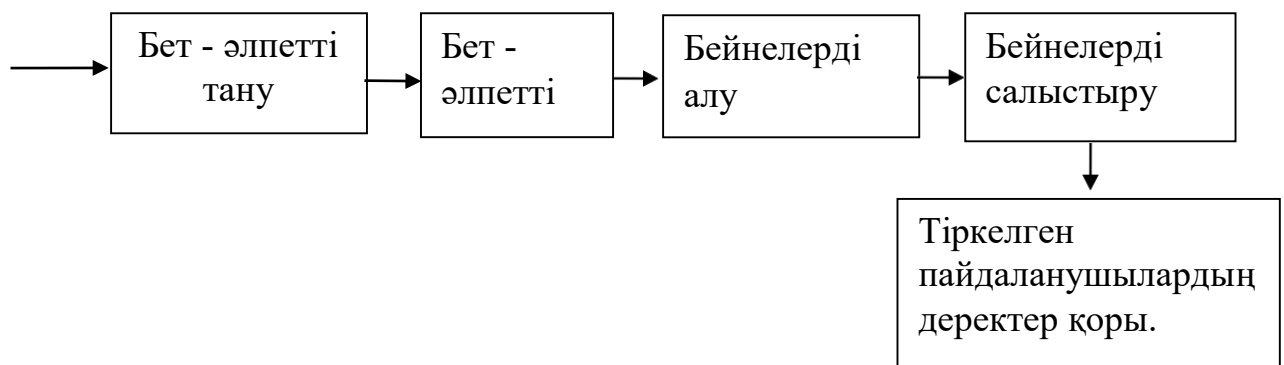
3 Бет әлпетті тану алгоритмі

3.1 Бетті танудың басты міндеттері

"Білтелерді қарастыру" функциясы бар бейнебақылау конструкциясы келесі өзара тәуелді функциялардың қайсысы деректер базасындағы қолжетімділікпен бейнелеуді салыстыру типі бойынша жұмыс істейді. :

- а) инициализация;
- б) жоспарлау;
- в) тексеру және мониторинг;
- г) сұранысты талдау;
- д) дизайн;
- е) кіші жобаларды әзірлеу және енгізу;
- ж) кіші жобаларды баптау;
- з) сілтемелерді бейнелеудің ақпараттық базасындағы беттерді қосу және жою;
- и) жалпы танылған тұлғалардың карточкаларын басып шығарыңыз және BMP Jed-файлды ұстаңыз;
- к) іздеуді тану және ақпараттық базада көрсету статистикасының басшылығы;
- л) бетті анықтау факторына сәйкес суретті тексеру;
- м) карточкаларды сәйкестендіру жүйелерінің стандарттарға арақатынасын, жоспардың маңызды және шешуші бағыттарынан кейінгі кернеуді тексеріңіз күш-жігердің беделі;
- н) тек жобаның сынамасына жүйелі итеративті сынау;
- о) жобаларды іске асырумен жеңудің қажеттіліктерін, модификациялары мен инциденттерін анықтау;
- п) дұрыс жүктелген жобаның қатысушылары.

Дипломдық жобадан кейінгі бағдарламалық ұсынымның Гамма-алгоритмі 6 суретке ұқсас.

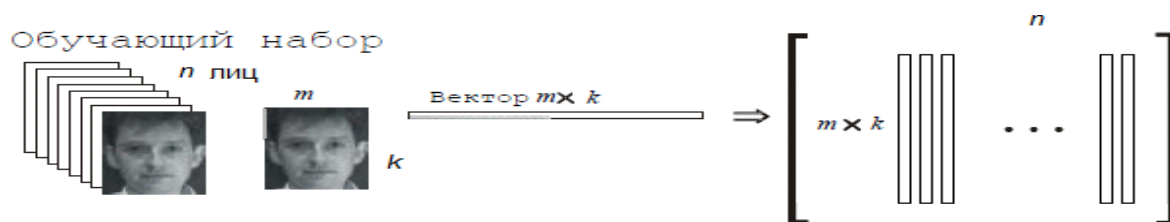


6 Сурет – Бағдарламалық қамтаманың алгоритімі

3.2 Бет кеңістігінің базисін құру

Бізде N портреттерін және X поливекторын тұтынамыз. Біз бірте-бірте бет орнын табамыз (қырлар кеңістігі, / шекара қашықтығы / = 100). Y_1 тірегінің бірінші поливекторы бар адамдарды одан да күшті "ажырату" болуы мүмкін. N_i $PNI=1$ $X_i - Y_1$ N . Space Y_1 Space көрсету қашықтығы және әрекетті қайталаңыз. Демек, біз Y_i тірегін жасаймыз, $1 = i = 100$ (сурет. 7).

Бізге $Imade$ Space орын z поливекторы ұсынылады. Оның жойылуын $facespas$ орын Базис. Бүгін біз осы вектор су коллекциясының қандай да бір тұлғасы дұрыс екенін анықтауға сәйкес келдік. Қалаған нәтижеге жету үшін K рецептін жоюға мүмкіндік береді.



7 Сурет – Бет кеңістігінің базисі

Бет әлпетті айырып тану технологиясы стандартты бейнекамерлермен жұмыс жасайды. Салыстыру үшін – бейне конференция үшін оңтайлы сапа секундтың 15 кадрдан бастап бейнеағымның жылдамдығын талап етеді. Бейне ағымның жоғары жылдамдығы сәйкестендірудің сапасын арттыруға мүмкіндік береді. Алыс арақашықтықта бет - әлпетті айырып тану кезінде, бейнекамера сапасы мен сәйкестендіру нәтижесі арасында тәуелділік пайда бастау. Стандартты дербес компьютерлерді қолдану кезінде дерекқордың көлемі 10000 кескіннен аспайды. Қазіргі таңда қолданылатын бет - әлпетті айырып тану әдістері бойынша қызықты және кеңінен қолдануға жақын, алайда осы әдістерге ғана жүгініп отыруымыз дұрыс болмайды. Ол еңуді бақылау жүйесінің көмекші ретінде тап-тырмайтын зат.

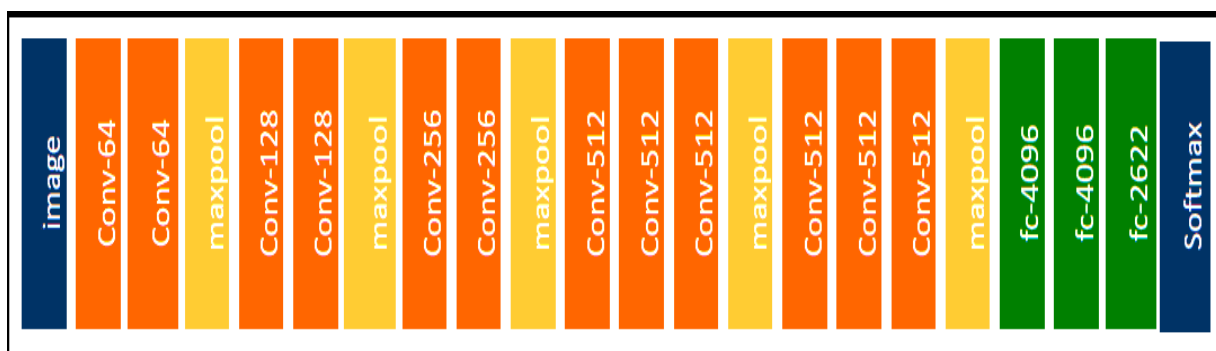
3.3 Абстракттілі қойылымы

Болашақта барлық фотосуреттер 100×100 пиксель өлшемі 256 сұр реңкі бар деп есептейміз. (8 - сурет) Мысалы, 40 фотосуреттен тұратын кейбір жаттығу кол- лекциясы бар (әр түрлі жағдайда 4 адамға 10 фотосурет).

Бірінші кезеңде алгоритм жалпы суреттен бейне бөлінеді. Содан кейін суретті қалпына келтіру бар. Суретті қалпына келтіруге келесі әрекеттер жатады:

- Сурет форматын 100×100 дейін өзгерту.
- 256 сұр реңкке түстерді түрлендіру.
- Суреттің жалпы жарықтығын кейбір орташа мәнге дейін өзгерту

Кейбір алгоритмдер үшін суреттегі бет мүмкіндігінше тік болуы қажет.



8 Сурет –VGG - Face моделі

Бұл жағдайда бетті дұрыс бұрышқа бұру қалыпқа келтіру кезінде жүзеге асырылады.

Тану алгоритмінің келесі кезеңі қолда бар тұлғаның сипаттамасын бөлу болып табылады. Таңдалған сипаттамалар тану алгоритміне қатты байланысты, сондықтан олардың мысалдары кейінірек келтірілетін болады. Сипаттамаларды бөлгеннен кейін сурет қажет емес. Танудың соңғы кезеңі қолда бар сипаттамалар бойынша есепке жауап беретін жіктеуішті қолдану болып табылады.

4 Бағдарлама жасау үшін пайдаланатын құралдар

4.1 Басты құралдардың негізгі сипаттамасы

Python бағдарламалау диалектісі-бұл тілді таңдаудың алғашқы себебі: ол машинамен жұмыс істеуге қандай да бір операциялық икемділікпен біріктірілмеген, бұл диалект осы қызметті орындау үшін қажетті бағдарламалаудың бас парадигмаларын ұстайды (ЖБП, көпфункционалды бағдарламалау, процедуралық бағдарламалау));

- еркін декламацияланатын бағдарламалау диалектісі;
- шамасы, пайдаланылмайтын кітапханалар.

NumPy кітапханашысы - үлкен кітапханамен бірлесе отырып, C, C++ және Fortran қосымшаларын қолдана отырып, жоғары деңгейлі және математикалық-математикалық функциялардың әсерлі кітапханасы, көп өлшемді массивтер Мен Python матрицаларының подмогасын қосады. Мүмкіндіктер:

- көп өлшемді массивтерге көмек (матрицаларды қоса алғанда));
- көп өлшемді массиві бар қызмет үшін жоғары деңгейлі нақты функцияларға көмек көрсету.

OpenCV кітапханашысы-компьютерлік көру, бейнелеу және жалпы мақсатты сандық алгоритмдер үшін бастапқы коды бар кітапхана. (EigenFaces, FisherFaces, LBPH);

Dlib кітапханашысы-автомобильдік оқыту алгоритмдері және әртүрлі қосымша құралдар кітапханасы. Бұл кітапханада ААМ (Active Appearance Model) дайындалған, көзқарас үшін жақсы. Оның негізгі қабілеті, әрине, жүктеме және нейрондық торларды іске қосу (inference). Сонымен қатар, модификация терең оқытудың кез-келген үш фреймворкасына-Caffe, TensorFlow Torch сиречь. Үш атақты фреймворк, біз бір қайталанбас фреймворкада барлығын түрлендіруге болмайды, бізге жүктелген модельдердің қызмет нәтижелерін қосуға мүмкіндік беріледі. Жүктеу кезінде Saff-те пайдаланушыға жақын маңдағы моральдық түрге арналған модификацияларды өзгерту жасалады.

OpenFace Кітапханашысы

Facenet әзірлеу-кітапхана facenet схемасына негізделген жетілдірілген анықтаудың нейрондық желісін пайдаланатын ашық бастапқы коды бар ыралдарды анықтау.

Кітапхана Keras-бұл Кітапхана ең аз операция санымен нейрондық түйіндерді тастайды. Яко түрлендіру нейрон байланыстарын қолданылады Sequential бейнелеу өнері модуль keras. Бұл модельде пайдаланылған кітапханалар.

Dense-пішінін Keras тапсыруымен модель. Біздің нейрондық сызық тығыз текстурадан (Dense) қалыптасады, барлық неврон келесі қабаттың барлық нейрондарымен біріктірілген.

Conv2d-жасалған екі өлшемді үзілу орамасын есептеу үшін қолданылады. Бұл функция матрица фигурасын қабылдайды және кез келген ұяшықты көбейтеді. Соңында, ол әрбір ұяшық бойынша өлшенген ақшаның

орташа қасиетін қайтарады.

Zeropadding2d-тензордың жоғарғы, төменгі, сол жақты және әділ үлесіндегі жолдар мен бағандарды қосу үшін жасалған сурет.

Maxpooling2d-пластикалық деректер қосылысының максималды әсері.

Flatten-бұл рецепт көптеген әдістермен жұмыс істейді, соның ішінде rows() және columns(). 2D-текстураны 1D-массив құрылымын қысқартады.

Dropout-қайта қарулануды алдын алуға көмектеседі оқыту кезінде кез-келген жаңарту туралы енгізу бірліктерінің бір ретсіз конструкциядан қалыптасады.

Activation -әрбір жаңарту кезінде 0 енгізу бірліктерінің кездейсоқ конструкциядан тұрады, бұл қайта жарактандыруды болдырмау көмектеседі.

PIL-PIL (Python көрсету кітапханасы) - Python суреттер графика қызметі үшін маманданған.

Мүмкіндіктер:

- екілік, жартылай тондық, индекстелген және толық түсті бейнелерге көмек;

- bmp, eps, EPS, GIF, JPEG, PDF форматтарына көмек. PNG, PNM, TIFF пішімде және басқа форматтарда оқылу және белгілеу;

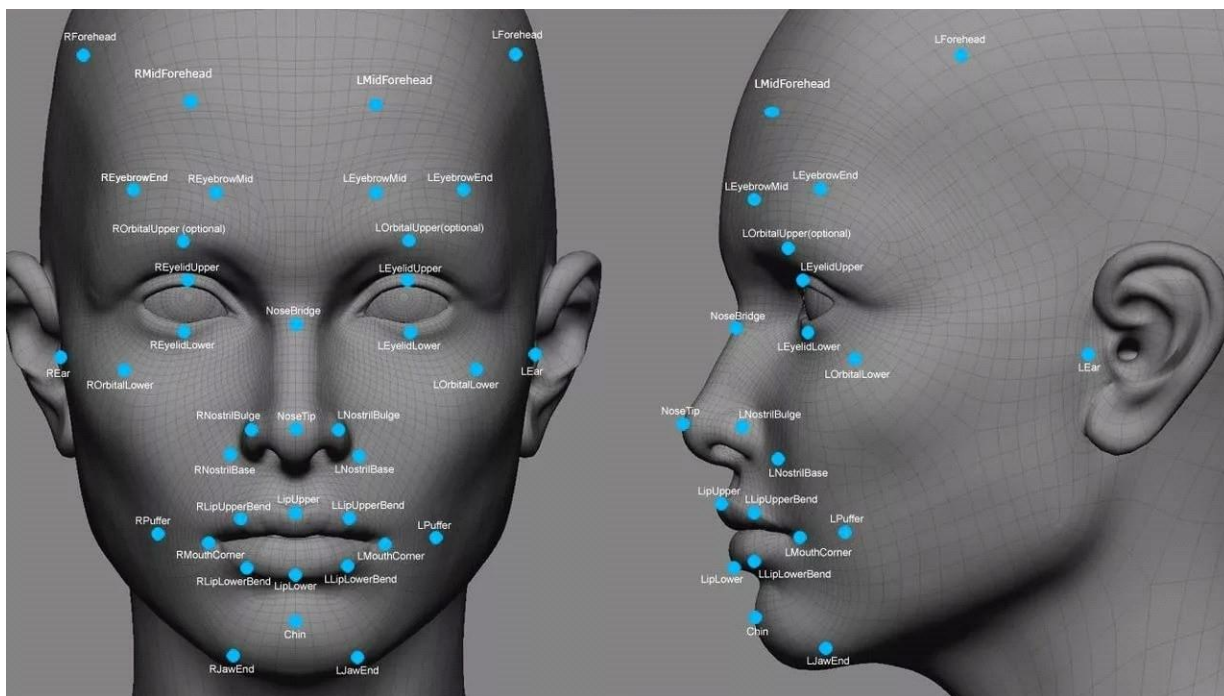
- көптеген форматтарды қолдану (ICO, MPEG, PCX, PSD, WMF және т. б. форматында) тек оқу үшін;

- бір пішімнен екінші пішімдегі бейнелердің өзгеруі;

- бейнелерді түзету (түрлі сүзгілерді пайдалану, масштабтау, сурет салу, матрицалық акциялар және т. б. –).

TensorFlow адам қабылдау қасиетіне жеткен бейнелерді көрсету және жүйелеу нысана бар нейрондық түйін жасау және жаттықтыру үшін Google бражасымен зерттелген автомобиль оқыту үшін табылған бағдарламалық кітапхана болып табылады. Зерттеу үшін сияқты, Google провианттарын әзірлеу үшін ұқсас. Кітапханамен жұмыс істеу үшін іргелі хор Python үшін сатылды және C++, Haskell, Java, Go және Swift үшін сату бар.

Бағдарлама құру кезінде, көп жағдайда, барлық адамдар мұрынның осындай биіктігін көзімен жеңеді. Өйткені сыртқы жағы үшін қандай тексеру нүктелерін атап өткен жөн, оның ішінде мұрын, көз, қас, ауыз, жақ, иек. Содан кейін мұндай сәйкестендіру үшін кейбір ағайын атқарушы тексеру нүктелерінде габариттік сүзгілердің қасиетін қысқартуға болады. Бос салаға қандай тексеру нүктелерін бөліңіз (сурет. 9). берілген адамның осы нүктелерінің қайда екенін анықтау үшін олардың орналасуы туралы қандай бейнелер болуы тиіс. Қалаған нәтижеге жету үшін "бірыңғай портрет" ұйымдастырамыз.



9 Сурет – Бақылау нүктелері

4.2 MVC архитектурасы

Осы қосымшаны әзірлеу үшін сәулет ретінде MVC архитектурасы қолданылады (Model-View-Controller, "Модельдік Ұсыныс-Контроллер"). MVC қолданба деректерін, пайдаланушы интерфейсін және басқару логикасын үш тәуелсіз компонентке бөледі: модель, ұсыну, контроллер.

Модель-деректерді ұсынады және бизнес-логиканы ұсынады. Сонымен қатар, бұл модельде мәліметтер базасы моделінің жай-күйін алады және сақтайды, сондай-ақ өзінің жай-күйін өзгертіп, контроллер командаларына жауап береді.

Көрініс-пайдаланушы интерфейсі. Модельдің өзгерістеріне жауап бере отырып, пайдаланушыға модельдің деректерін көрсетуге жауап береді.

Контроллер-пайдаланушылардың сұрауларын өңдейді. Әдетте пайдаланушы интерфейспен өзара әрекеттеседі, және, өз кезегінде, URL сұралады. Сұрау контроллермен өңделеді. Контроллер модель деректерін көрсетеді және жауап ретінде view көрсетеді.

MVC үлкен кодты бөлікке бөлу арқылы жеңілдетеді. Ұзын кодты жазу итеріп, қатесіз өзгерту қиын.

MVC белгілі бір бағдарламалау тіліне тәуелді емес және объектілі-бағытталған әдісті немесе басқа да мысалдарды талап етпейді.

Мұнда бөлу model, view және controller аттары бар үш файл (немесе үш файл бар қалталар) қажет дегенді білдірмейді. Практикада модель қосымшалардың негізгі көлемін алады және әртүрлі кластар - мәндер, сервистер, ДБ-мен жұмыс кластары түрінде ұсынылған және сыныптардың әрбір түрі үшін жеке папкалар жасалады.

MVC серверлік веб-қосымшаларды және үстелдік (клиенттік) қосымшаларды қоса алғанда, көптеген түрлі қосымшаларға қолданылуы мүмкін. Олардың арасындағы айырмашылық веб - бағдарлама пайдаланушының сұрауына жауап ретінде өңделеді, нәтижені көрсетеді (әдетте, веб-бет) және

жұмысты аяқтайды. Басқа сұраулар түскенде, бұл өңдеу үшін жаңа тәуелсіз бағдарлама көшірмесі орындалады. Веб-қосымшалардан, дербес компьютерлерден, мобильді қосымшалардан (және JavaScript-та жасалған браузер беттерін пайдаланатын қосымшалардан) айырмашылығы, олардың ұзақ қызмет ету мерзімі бар. Олар пайдаланушының әр түрлі сұрауларын өңдейді және шығусыз экрандағы ақпаратты жаңартады.

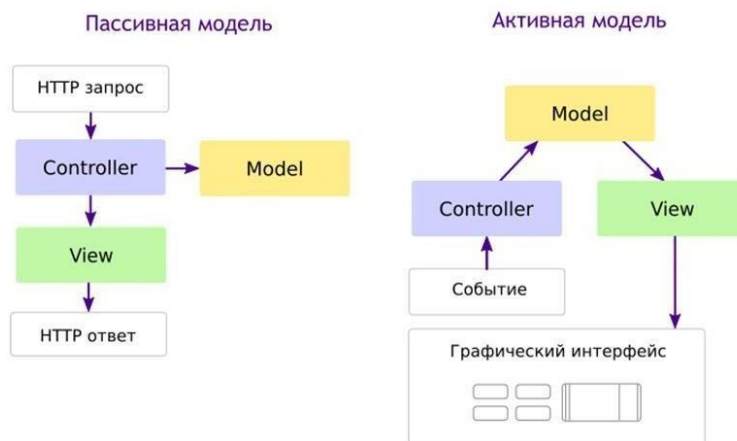
MVC құрамдастары арасындағы өзара әрекеттесу серверлік және десктопты қосымшаларда сәл таңқаларлық іске асырылған, веб-бағдарлама пайдаланушының бір сұранымын өңдейді және аяқталады, ал десктопты қосымша қайта іске қосусыз көп сұраныстарды өңдейді.

Серверлік қосымшалар әдетте "пассивті" модельді, ал десктопты қосымшалар – "белсенді" модельді қолданады. Белсенді үлгілерге қарағанда, пассивті модельдер өзгерістер туралы ескертулерге қол қоюға және алуға мүмкіндік береді. Серверлік қосымшалар қажет емес.

Белсенді модельді схемада түр модельдегі өзгерістерге қол қойылады. Содан кейін оқиға болған кезде (мысалы, пайдаланушы батырманы басады) контроллер шақырылады. Ол модель үшін деректерді өзгерту пәрмендерін ұсынады. Модель өзінің жазылушыларына (оның ішінде түрі) деректер өзгергенін хабарлайды және түрі бағдарламаның интерфейсін жаңартады.

Белсенді модельдері бар схемалар ұзақ уақытқа есептелген. Керісінше, пассивті модель схемалары әдетте қысқа мерзімді қосымшаларда қолданылады.

Серверлік қосымшалар пассивті модельдік жүйелерді пайдаланады. Пайдаланушы форум бетіне кіреді деп болжаймыз. Браузерде ол http-хабарлама тізімі бар бетті іздеу сұрауын жібереді. Осы уақытта контроллер іске қосылады, пайдаланушы сұрауын талдайды және Model-ден хабарламалар тізімін сұрайды. Оны алып, ол View шақырады, тізімді жібереді және оны веб-бет түрінде көрсетеді. Бұл жағдайда, пайдаланушы нысанды толтырады, әрі қарай деректерді қабылдайтын осы формадағы деректерді өңдеуге жауап беретін контроллер шақырылады және жаңа хабарламаны тексеру және дерекқорға енгізу үшін модельді іске қосады, содан кейін хабарламаларды қарау бетіне редиректімен HTTP жауабын қайтарады. 10-суретте компоненттердің өзара әрекеттесуін бейнелейтін схема келтірілген.



10 Сурет -MVC компоненттерінің өзара әрекеттесуі

Django архитектурасы "модель-көрініс-Контроллер" (MVC) сияқты. MVC классикалық моделінің контроллері Django-да көрініс (View) деп аталатын деңгейге сәйкес келеді, ал таныстыру логикасы Django-да Шаблондар деңгейінде (Templates) жүзеге асырылады. Осыған байланысты Django деңгейлік архитектурасын жиі "үлгі-үлгі-көрініс" (MTV) деп атайды.

Бастапқыда Django-ның әзірлеуі жаңалық ресурстарымен неғұрлым ыңғайлы жұмысты қамтамасыз ету үшін жүргізілді, бұл архитектураға қатты әсер етті: фреймворк ақпараттық сипаттағы веб-сайттарды жылдам әзірлеуге көмектесетін бірқатар құралдарды ұсынады. Мысалы, әзірлеуші сайттың әкімшілік бөлігіне арналған контроллерлер мен беттерді жасауды талап етпейді, Django-да Django-да жасалған кез келген сайтқа қосуға болатын мазмұнды басқару үшін кірістірілген бағдарлама бар және бірден бірнеше басқаруға болады бір серверде сайттар. Әкімшілік қосымша барлық жасалған әрекеттерді протоколдай отырып, сайт толтырудың кез келген нысандарын жасауға, өзгертуге және жоюға мүмкіндік береді және пайдаланушылар мен топтарды басқару үшін интерфейсті ұсынады (әр объект бойынша құқықтармен). Веб фреймворк Django Instagram, Disqus, Mozilla, The Washington Times, Pinterest сияқты ірі және танымал сайттарда қолданылады, lamoda және т. б.

4.3 Қолданбалы бағдарламалық қамтамасыз ету

4.3.1 Google Chrome Браузері

Google Chrome-тегін браузер мен Blink қозғалтқышы негізінде Google жасаған браузер. Бірінші нұсқа 2008 жылдың 11 желтоқсанында шығарылды. StatCounter мәліметтері бойынша, Chrome - әлемдегі ең танымал браузер.

Secure Chrome жаңартуларды үнемі алады (зиянды коды бар сайттар) және пайдаланушы зиянды бетке келуге тырысқанда ЕСКЕРТЕДІ.

Плагиндер (ең жиі қолданылатын Adobe Flash Player) браузердің бөлігі емес және құмға қосылуы мүмкін емес. Осындай бет кодын қауіпсіз орындау операциялық жүйесі Қосымша қорғалмаған плагиндермен өшіріледі.

2010 жылдың 30 наурызында Adobe Flash браузерге қосылады және пайдаланушылардан оны жүктеу мен орнатуды талап етпейді. Flash Player автоматты түрде жаңартылады.

Adobe Flash Player 5.0 нұсқасымен біріктірілген.

4.2 нұсқасынан бастап, Chrome NPAPI плагиніне қызмет көрсетуді тоқтатты. Бұл плагин жаңа PPAPI пішімін (Pepper API) қолдайды, ол жеке процесте (құм) плагинді іске қосу үшін арнайы әзірленген. Осылайша, бұл осалдық нұсқасы плагиндегі зиянды кодты жүйеге енгізе алмайды.

Браузер құпия режимді қолдайды. Терезеде көрінетін беттер "Инкогнито" браузер тарихында немесе іздеу тарихында көрсетілмейді және cookie файлдары сияқты компьютеріңізде басқа із қалдырмайды. Осы терезе жабылғаннан кейін пайдаланушылар автоматты түрде жойылады. Дегенмен, барлық жүктелген файлдар немесе жасалған бетбелгілер сақталады.

2015 жылдың көктемінде браузер ЕСКЕРТЕДІ, "құпия режимге" өту басқа пайдаланушылардың, серверлердің немесе бағдарламалардың жұмысына әсер

етпейді және келесілерге кедергі болмайды:

- ақпаратты жинайтын немесе оны басқаларға хабарлайтын интернет-сайттар;

- веб-сайттарға кіруді қадағалайтын интернет провайдерлер мен қызметкерлер;

- пернелерді басуды қадағалайтын зиянды бағдарламалық қамтамасыз ету.

Speed Chrome жоғары өнімді JavaScript V8 өңдегішін пайдаланады. Сонымен қатар, Chrome бетті жүктеуді жеделдету үшін DNS алдын ала таңдау функциясын пайдалана алады.

Сенімділік. Chrome көптеген процестердің архитектурасын пайдаланады және жиі әрбір карта немесе плагин жеке процесс бар. Бұл процедура бетбелгілерге араласуды болдырмайтын оқшаулау деп аталады.

Chrome тапсырмаларды пайдаланушы жадын, процессорлар мен онлайн арналарды пайдаланатын сайттар мен плагиндерді көре алатын тапсырмаларды реттеушінің қажеттіліктеріне сәйкес орындай алады.

Әрбір карточканың жоғарғы жағында омнибокстың адрестік жолы. Бұл мекенжай жолы мен іздеу жолдары комбинациясы. Егер омнибокс мекен-жайы URL жазуына сәйкес келмесе, ол іздеу сұрауын іздеу жүйесіне қайта жібереді, мысалы, мекенжайдан жіберілген мекенжайдың басында орын болмаса. Авто толтыру, сондай-ақ бетбелгілерді және бұрын кірген беттердің тарихын іздеуге көмектеседі. Браузер түрлі іздеу жүйелерін адрес жолынан тікелей тудыруы мүмкін.

Бағдарлама браузерлердің бірі арқылы іске қосылады (Internet Explorer, Opera, Firefox, Google Chrome) <http://localhost/8080>.

4.4 Аспаптық бағдарламалық қамтамасыз ету

4.4.1 PyCharm әзірлеу ортасы

PyCharm-Python бағдарламалау тілі үшін интеграцияланған даму ортасы. Кодты талдау құралы, графикалық реттеуші, юнит-тестерді іске қосу құралы және Django веб-әзірлемесін қолдайды. PyCharm IntelliJ IDEA негізінде JetBrains компаниясымен жасалған. PyCharm Windows, Mac OS X және Linux операциялық жүйелерімен жұмыс істейді. Негізгі мүмкіндіктер:

- Кодты статистикалық талдау, синтаксис және қателерді жарықтандыру.

- Жоба және бастапқы код бойынша навигация: жобаның файлдық құрылымын көрсету, файлдар, сыныптар, әдістер мен әдістерді пайдалану арасында жылдам өту.

- Рефакторинг: атауын өзгерту, әдісті шығару, айнымалыны енгізу, константаны енгізу, әдісті көтеру және түсіру және т. б.

- Django фреймворк пайдалану веб-әзірлеу құралдары.

- Python үшін кірістірілген реттеуші.

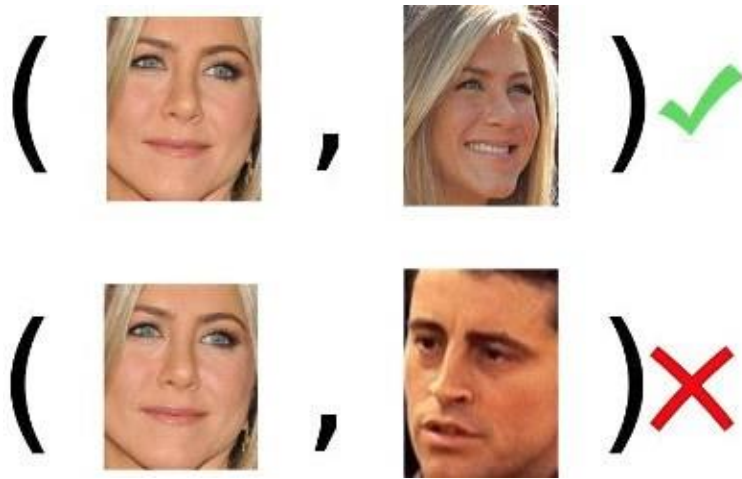
- Юнит-тестілеу үшін кіріктірілген құралдар.

- Google App Engine пайдаланып әзірлеу.

- Нұсқаларды бақылау жүйесін қолдау: Mercurial, Git, Subversion, Perforce және CVS үшін жалпы пайдаланушы интерфейсі.

4.5 Бағдарламалық қамтаманың көрсетілімі

Осы дипломдық жоспарда келесі сұрақтар қаралды: 1. Берілген дипломдық жұмыста қолданылатын негізгі ұғымдар; 2. Берілген дипломдық жұмыста қолданылатын негізгі ұғымдар. Адамды ажырату үшін оны соңғы шекарада бір рет белгілеу керек. Дәлірек айтқанда, шеткі шекарада бір фотокарточканы орналастыру үшін оны қандай да бір бейнеге ұқсайды. Кейде біз бізге жаңа, сөзсіз фотосуретті көрсеттік, біз оны барлық есте сақтау карточкаларымен салыстырамыз және жауап бере аламыз: біз осы ағайын көрдік және оны анықтай аламыз, немесе біз осы күйеуіммен кездеспедік. Демек, жоғарыда аталған проблема келесілерге негізделеді: екі карточка (p_1 ; p_2) бар, олардың адамға есептелгенін немесе қосылмағанын анықтау. (Сур. 11) басқаша айтқанда, олар бір адамға ажырамас қасиеттерінің бар-жоғын анықтайды.



11 сурет – Суреттерді салыстыру

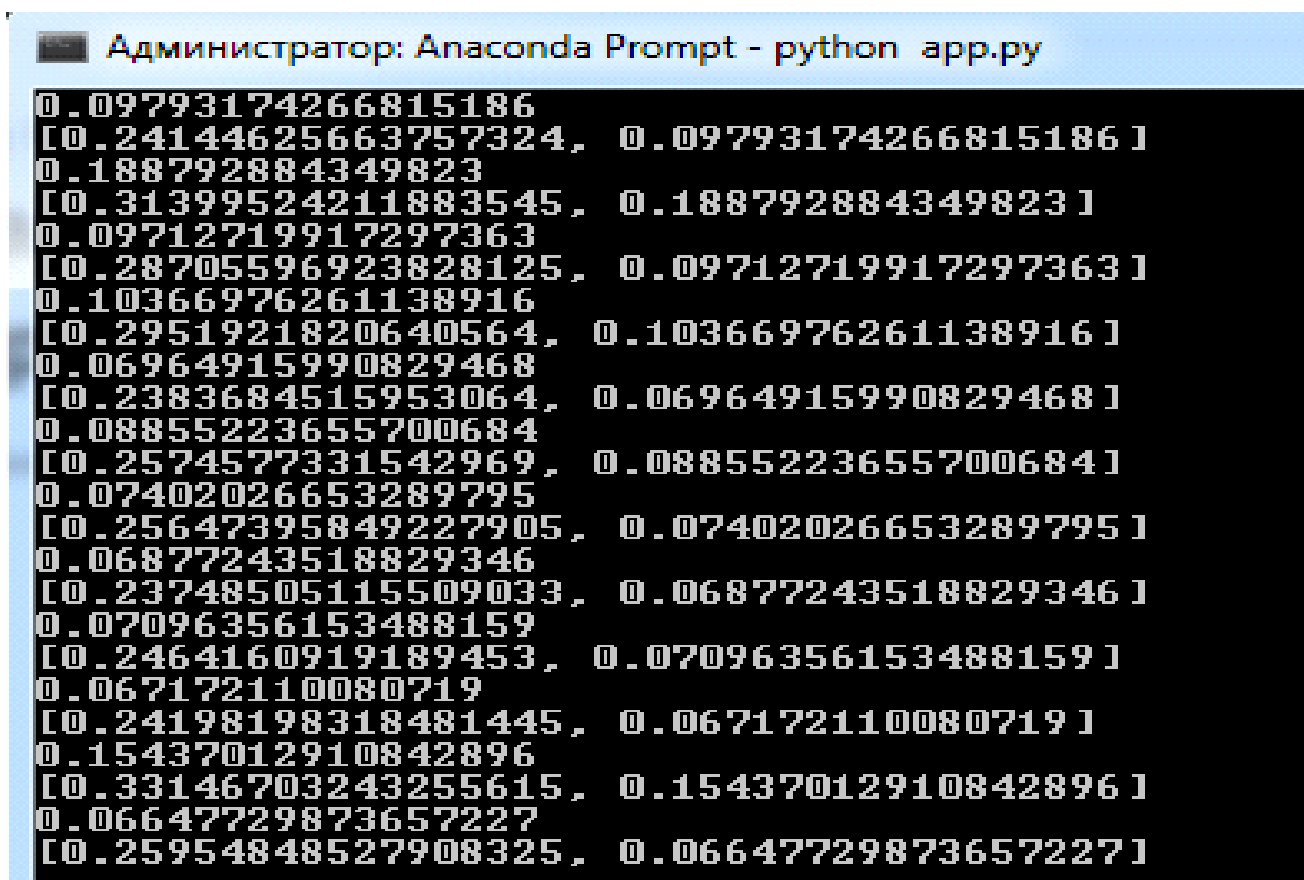
"Үйретеді" коды белгілі бір тұлғаны теңдестіру. Шағын видеокамера және басқа құрылғылар тіркеу кезеңінде тұлғаның сандық бейнеленуін жасау үшін фигураны сканерлейді. Беттерді тану 20-30 секунд созылады, сайып келгенде, суреттер дұрыс ұйымдастырылады. Егер бұрын камера ешкім жоқ болса, телекамера жабылады. Ең дұрысы, бейнелеу деректері бірнеше түрлі ракурстар мен бет бар. Заттардың мұндай жағдайы атқарушылық ақпаратты қадағалауға рұқсат етеді. Бұл ретте, Қазақстан Республикасының Білім туралы заңнамасының талаптарына сәйкес, қазіргі уақытта кадрларды даярлау мен қайта даярлаудан және кадрлардың біліктілігін арттырудан кейін жыртылған. Мысалы, көздің жоғарғы формочерленуі, ауыздың шеті мен бет аймағы. Биометриялық технологияларға арналған кейбір Алгоритмдер бет шашының ықтимал өзгерістерін ойлауға рұқсат береді, олар шаш ұзындығынан артық Көлемді талдау үшін пайдаланылмайды. Кез келген юзердің бейнелеу модификациясы биометриялық жүйенің ақпараттық базасында жинақ болып табылады. Кейде личиканың көрінісі 100x100 мөлшерінде орналасады, телекамера оны итеріп, оны 224x224 фигураға жақындатады. Үлгі қауіпсіз дигиталды пішімге ауыстырылады. Биометриялық жүйелердің жанында сәйкестендіру туралы қаулы, тек бетінен кейін ғана және әділ биометриялық сипаттамалардың алғы жағдайында, кейде оң

болып көрінеді. Заттардың мұндай жағдайы, біріншіден, көптеген биометриялық сипаттамалардың өзгеруімен тікелей біріктірілген. Конструкция қате ықтималдығының негізделген деңгейі бар. Егер камераны қарап шығу алдында 0, 16-дан көп көріністі таниды (12-сурет).

Сонымен қатар, әр түрлі технологияларды қолдану қате болуы мүмкін. Енуді бақылау жүйелері үшін, биометриялық технологияларды қолдану кезінде, бастапқыда анықтауды, не көп мәнді "тосқауыл" "шығармауды немесе" өз адамын" күтуді таңдайды. Биометриялық технологияларды пайдаланушылар үшін зиянсыздық жүйелеріндегі ең үлкен сәт қолдану қиындылығы болып табылады. Сканерленетін личико ешқандай ыңғайсыздық сезе алмайды. Бұл жағдайда, әрине, адам бойынша анықтаманы әзірлеу тек қызықты әдіс болып табылады.

Жалпы қабылданған ұсынылған тілектің арақатынасын анықтау үшін достас деректер қоймасында сақталатын үлгімен теңестіріледі. Деңгейі ұқсастық, настоящий үшін тексеру ұсынылады бір табалдырығының және может статья реттелуі үшін күш-ДК, медли тәуліктің әр түрлі бейнелерді персоналдың және өзге де факторлар. Идентификация анықтау, идентификациялау, тану сиретуін анықтау нұсқасында жүргізілуі мүмкін. Анықтау кезінде ақпараттық базада сақталатын модельдің және қабылданған деректердің өтемақысы куәландырылған болады.

Идентификация-деректер қоймасында сақталған үлгілердің негізгі Жаңа айқындалған кадрының жанында камера бұрғысы деп аталады.



```
Администратор: Anaconda Prompt - python app.py
0.09793174266815186
[0.24144625663757324, 0.09793174266815186 ]
0.188792884349823
[0.31399524211883545, 0.188792884349823 ]
0.09712719917297363
[0.28705596923828125, 0.09712719917297363 ]
0.10366976261138916
[0.2951921820640564, 0.10366976261138916 ]
0.06964915990829468
[0.2383684515953064, 0.06964915990829468 ]
0.08855223655700684
[0.2574577331542969, 0.08855223655700684 ]
0.07402026653289795
[0.25647395849227905, 0.07402026653289795 ]
0.06877243518829346
[0.23748505115509033, 0.06877243518829346 ]
0.07096356153488159
[0.2464160919189453, 0.07096356153488159 ]
0.067172110080719
[0.24198198318481445, 0.067172110080719 ]
0.15437012910842896
[0.33146703243255615, 0.15437012910842896 ]
0.06647729873657227
[0.25954848527908325, 0.06647729873657227 ]
```

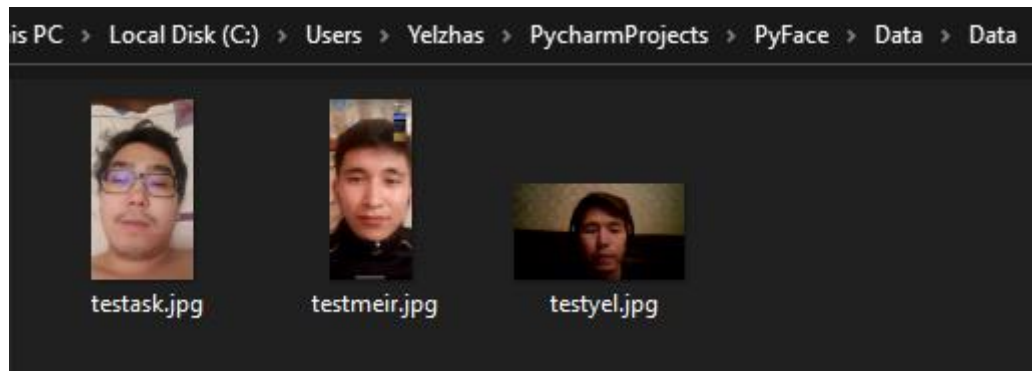
12 Сурет – ықтималдылығының деңгейі

Бағдарлама бет - әлпетті суретке тусіріп, оны базаға тіркейді. (13 - сурет)

Бет әлпетті суретке айналдырып оны қырқып dataset папкасына жибереді. (14 - сурет)

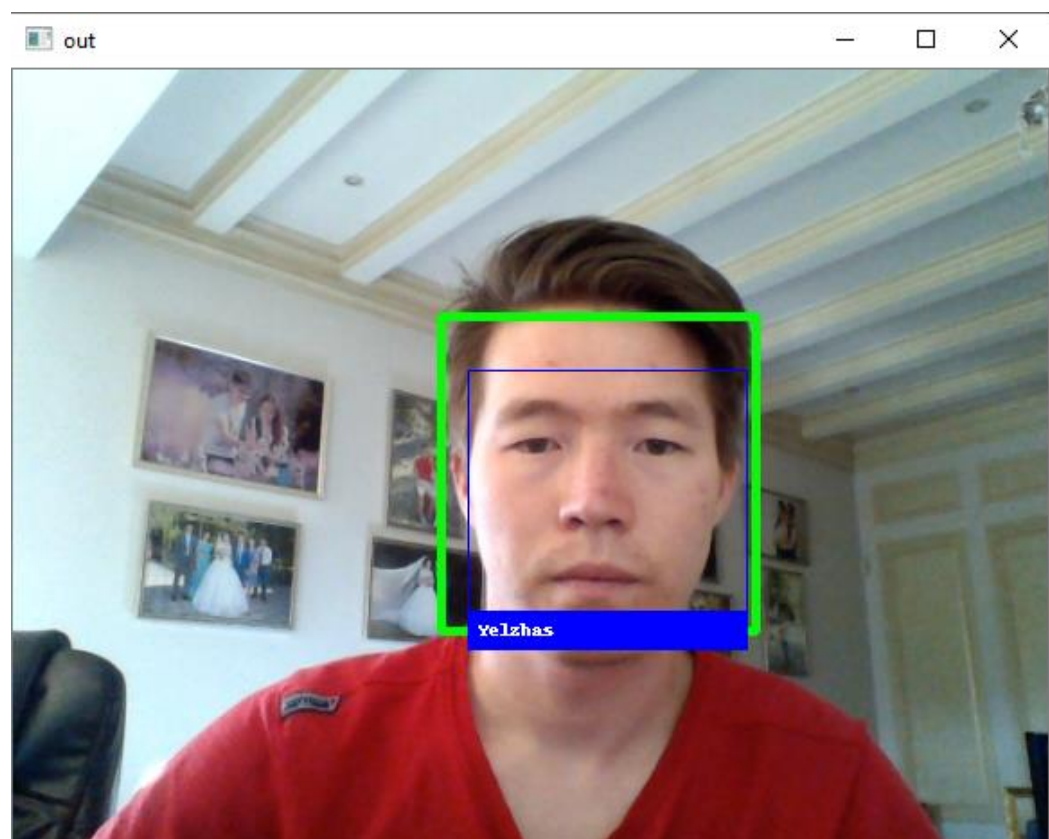
```
C:\Users\Yelzhas\PycharmProjects\PyFace\venv\Scripts\python.exe C:/Users/Yelzhas/PycharmProjects/PyFace/PyFaceApp.py
<PIL.Image.Image image mode=RGB size=576x432 at 0x18F0F4C0390>
['Yelzhas']
<PIL.Image.Image image mode=RGB size=576x432 at 0x18F0F4C03C8>
['Yelzhas', 'Askar']
```

13 Сурет – Суретті базаға тіркеу

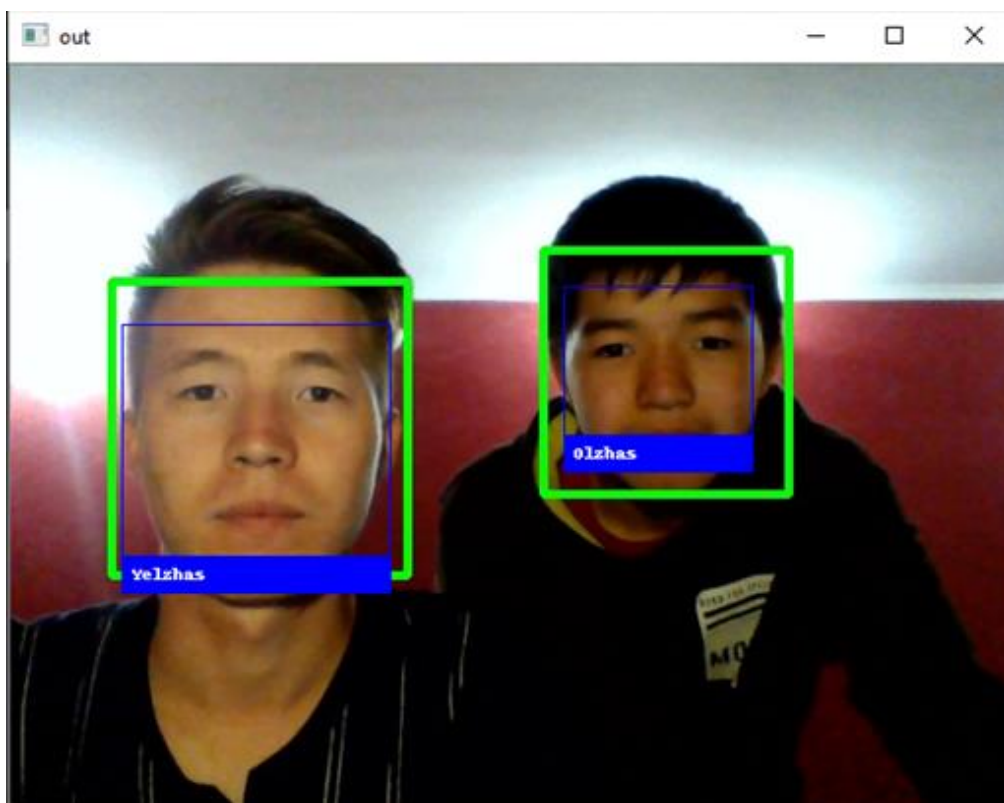


14 Сурет – data папкасы

Бейнені сурет ретінде алып оны камера арқылы 0 адресімен вебке шығарып береді. Ары қарай бет - әлпетті тану жүреді (15 - сурет).



15 Сурет - Бет – әлпетті тану



15 Сурет – Бірнеше тұлғаны тану

Бетті танудың бағдарламасын құрудағы ең басты мақсаты бұл бірнеше адамды бір уақытта тану болып табылады. Мұнда, суперкомпьютер көмегімен 30 бетті бір уақытта бір секундта анықтап шыға алады. Ол бір адамның бет – бейнесін бір тусіргенде бірнеше сурет алып салыстырады дағы қайсы сурет тұнық ,яғни, качестваы жоғары соны алады да салыстырады.

ҚОРЫТЫНДЫ

Мен осы жұмыста беттерді анықтаудың үйлесімділігін жасадым. Бетті қарау-бұл сыртқы технология арқылы жапырақтарды анықтау тәртібі. Білікшелерді анықтау конструкциясы биометрияны білікшелердің және бейнелердің шағылысуы үшін кәдеге жаратады. Ол деректерді іздеу үшін танымал тұлғалардың базаларымен салыстырылады. Түлкілерді қарау Жеке сәйкестендіруді тексеруге қосалқы, бірақ ол құпиялылық мәселелерін қозғайды. Личико ағасы біздің өзара қарым-қатынасымызда ұлы мағынада ойнайды, адамдар фигурасын қандай итермелейді. Қорғаныс биометриялық әзірленуі яко кілтінің личиктері пайдаланыла отырып, ағасы қайталанбас, соңғы екі жылда құқық қорғау органдарында ұқсас кеңістікті пайдалану қабілеттілігіне үлкен қызығушылықты бөлді.

Бағдарлама адамның осы бетіндегі учаскенің 80 нүктесін сипаттайды. Осы басты нүктелерді назарға ала отырып, адам жазықтығының тұрақсыз мәндерін өлшеу үшін соңғы нүктелер ұсынылады, оның ішінде мұрынның ұзындығы мен ауқымдылығы, розеткалардың күрделілігі бетін қарау және пішу. Конструкция ағынның жазықтығын сандық бейнелеуден кейін тораптық нүктелер үшін берілген шақыру және алынған яко жеңіл мағыналы ізді сақтау арқылы жұмыс істейді. Беттердің арабескі кейіннен бейнелеуді салыстыру үшін база сипатында немесе тұлғалар арқылы алынған деректермен қолданылады.

Беттерді анықтаудың конструкциясы тек 80 басты нүктесін ғана кәдеге жаратады, бірақ бұл мақсатты тұлғаларды сөзсіз алдын ала айқындай алады, кейде бұл конъюнктура қолайлы. Алайда, егер субъектінің саласы алдын ала қарастырусыз орналастырылады, жартылай қараңғы немесе профильде сәл жарық түсіреді, Бағдарламаның бұл сипаты сенімді болады. Мемлекеттік стереотиптер мен технологиялар институтының (NIST) нақты мәліметтері бойынша, 1993 жылдан бастап личикті анықтау жүйесіндегі жалған-позитивті жағдайлардың радиожілік жасы екі жас ішінде екі есе азайды. Босануға конструкциясы анықтау беттерді ұсынылады қолайлы.

ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

- 1 Jain A.K., Bolle R., Pankanti S., eds., Biometrics: Personal Identification in Networked Society, Kluwer Academic, Norwell, Mass., 1999.
- 2 Brin D., Transparent Society: Will Technology Force Us to Choose Between Privacy and Freedom? Addison-Wesley, New York, 1998.
- 3 Айзерман М.А., Браверман Э.М., Розоноэр Л.И. Метод потенциальных функций в теории обучения машин. - М.: Наука, 2004. - 384 с.
- 4 Горбань А., Россиев Д. Нейронные сети на персональном компьютере. //Ново- сибирск, Наука, 1996. – С 114 – 119.
- 5 Мазуров В.Д. Комитеты систем неравенств и задача распознавания // Киберне- тика, 2004, № 2. С. 140-146.
- 6 Потапов А.С. Распознавание образов и машинное восприятие. - С-Пб.: Политех- ника, 2007. - 548 с
- 7 Растрингин Л. А., Эренштейн Р. Х. Метод коллективного распознавания. 79 с. ил. 20 см., М. Энергоиздат, 2006. – 80 с.
- 8 Рудаков К.В. Об алгебраической теории универсальных и локальных ограниче- ний для задач классификации // Распознавание, классификация, прогноз. Матема- тические методы и их применение. Вып. 1. - М.: Наука, 2007. - С. 176-200.
- 9 <https://ru.wikipedia.org/>
- 10 <https://www.itv.ru/products/intellect/faceintellect/>
- 11 <https://securityrussia.com/blog/face-recognition.html>
- 12 Jones, M., Viola, P. (2001) Robust Real-Time Face Detection. URL: International Journal of Computer Vision, 57(2), 137-154, 2004. Available from: [http://www.vi- sion.caltech. edu/html-files/EE148-2005-Spring/pprs/viola04ijcv.pdf](http://www.vision.caltech.edu/html-files/EE148-2005-Spring/pprs/viola04ijcv.pdf) (ссылка проверена 28 мая 2013)
- 13 Turk, M., Pentland, A. (1991) Face Recognition Using Eigenfaces URL: <http://www.cs.ucsb.edu/~mturk/Papers/mturk-CVPR91.pdf> (ссылка проверена 28 мая 2013)
- 14 A novel approach for human face detection from color images under complex back- ground <http://visgraph.cs.ust.hk/biometrics/Papers/Face/pr2001-10-1.pdf>(ссылка прове- рена 28 мая 2013)

А қосымшасы
(міндетті)
Бағдарламалық код үлгісі

PyFaceApp.py – бет – әлпетті камера арқылы тану

```
import face_recognition
import numpy as np
from PIL import Image, ImageDraw
from IPython.display import display
import numpy
import cv2
import requests

face_cascade =
cv2.CascadeClassifier('./Data/cascade_files/haarcascade_frontalface_alt.xml')
video_capture = cv2.VideoCapture(0)
scaling_factor = 0.9

image1 = face_recognition.load_image_file("Data/testmeir.jpg")
image1_face_encoding = face_recognition.face_encodings(image1)[0]
image2 = face_recognition.load_image_file("Data/testyel.jpg")
image2_face_encoding = face_recognition.face_encodings(image2)[0]
image3 = face_recognition.load_image_file("Data/testask.jpg")
image3_face_encoding = face_recognition.face_encodings(image3)[0]

# Create arrays of known face encodings and their names
known_face_encodings = [
    image1_face_encoding,
    image2_face_encoding,
    image3_face_encoding
]
known_face_names = [
    "Meirman",
    "Yelzhas",
    "Askar"
]

def detec_faces():
    list_of_attendance = []
    # <----->
```

А қосымшасының жалғасы

```
unknown_image = face_recognition.load_image_file("Data/out_img.jpg")

face_locations = face_recognition.face_locations(unknown_image)
face_encodings = face_recognition.face_encodings(unknown_image,
face_locations)

pil_image = Image.fromarray(unknown_image)

draw = ImageDraw.Draw(pil_image)

for (top, right, bottom, left), face_encoding in zip(face_locations, face_encodings):
    matches = face_recognition.compare_faces(known_face_encodings,
face_encoding)

    name = "Unknown"

    face_distances = face_recognition.face_distance(known_face_encodings,
face_encoding)
    best_match_index = np.argmin(face_distances)
    if matches[best_match_index]:
        name = known_face_names[best_match_index]
        list_of_attendance.append(name)

    draw.rectangle(((left, top), (right, bottom)), outline=(0, 0, 255))

    text_width, text_height = draw.textsize(name)
    draw.rectangle(((left, bottom - text_height - 10), (right, bottom)), fill=(0, 0, 255),
outline=(0, 0, 255))
    draw.text((left + 6, bottom - text_height - 5), name, fill=(255, 255, 255, 255))

del draw

display(pil_image)

open_cv_image = numpy.array(pil_image)

o_image = cv2.cvtColor(open_cv_image, cv2.COLOR_BGR2RGB)

cv2.imshow("out", o_image)
print(list_of_attendance)

# <-----REQUEST----->

API_ENDPOINT = "https://obscure-sands-56819.herokuapp.com/opencv/"
```

А қосымшасының жалғасы

```
s = "DDD "

for i in list_of_attendance:
    s += i + " "

data = {'lists': s,
        'api_paste_format': 'python'}

# sending post request and saving response as response object
r = requests.post(url=API_ENDPOINT, data=data)

# extracting response text
pastebin_url = r.text

# <-----END---REQUEST----->

cv2.waitKey(0)

# <----->

#detec_faces()

while True:
    ret, frame = video_capture.read()

    rgb_frame = frame[:, :, ::-1]

    frame = cv2.resize(frame, None, fx=scaling_factor,
                       fy=scaling_factor, interpolation=cv2.INTER_AREA)

    face_rects = face_cascade.detectMultiScale(frame, scaleFactor=1.3,
minNeighbors=3)

    for (x, y, w, h) in face_rects:
        cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 3)

    cv2.imshow('Video', frame)

    key = cv2.waitKey(10)

    if key == 27:
```

А қосымшасының жалғасы

```
break
    elif key == ord('s'):
        cv2.imwrite("Data/out_img.jpg", frame)
        detec_faces()
```

FaceDetect.py – суретке түсіру жүйесі

```
import cv2
import numpy as np

face_cascade = cv2.CascadeClassifier('./cascade_files/haarcascade_frontalface_alt.xml')

cap = cv2.VideoCapture(0)
scaling_factor = 0.9

while True:
    ret, frame = cap.read()

    center = 0
    scale = 1.0

    frame = cv2.resize(frame, None, fx=scaling_factor,
                       fy=scaling_factor, interpolation=cv2.INTER_AREA)

    face_rects = face_cascade.detectMultiScale(frame, scaleFactor=1.3,
minNeighbors=3)

    # print(face_rects)

    for (x, y, w, h) in face_rects:
        cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 3)

    cv2.imshow('Face Detector', frame)

    c = cv2.waitKey(1)
    if c == 27:
        break

cap.release()
cv2.destroyAllWindows()
```

А қосымшасының жалғасы

dlib_facial_landmarks.py – суреттен бетті тану

```
from imutils import face_utils
import dlib
import cv2

# Vamos inicializar um detector de faces (HOG) para então
# let's go code an faces detector(HOG) and after detect the
# landmarks on this detected face

# "shape_predictor_68_face_landmarks.dat" = our pre-trained model directory, on my
# case, it's on the same script's diretory.
detector = dlib.get_frontal_face_detector()
predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")

cap = cv2.VideoCapture(0)

while True:
    # Getting out image by webcam
    _, image = cap.read()
    # Converting the image to gray scale
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

    # Get faces into webcam's image
    rects = detector(gray, 0)

    # For each detected face, find the landmark.
    for (i, rect) in enumerate(rects):
        # Make the prediction and transform it to numpy array
        shape = predictor(gray, rect)
        shape = face_utils.shape_to_np(shape)

        # Draw on our image, all the finded cordinate points (x,y)
```

А қосымшасының жалғасы

```
for (x, y) in shape:
    cv2.circle(image, (x, y), 2, (0, 255, 0), -1)

# Show the image
cv2.imshow("Output", image)

k = cv2.waitKey(5) & 0xFF
if k == 27:
    break

cv2.destroyAllWindows()
cap.release()
```

Б қосымшасы
(міндетті)
Бағдарламалық код үлгісі

Manage.py – бет – әлпетті камера арқылы тану

```
#!/usr/bin/env python
import os
import sys

if __name__ == '__main__':
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'prepod.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)
```

Models.py - Django – база данных

```
from django.db import models
from django.contrib.auth.models import AbstractUser
from django.utils.translation import ugettext_lazy as _
from django.contrib.postgres.fields import ArrayField
from django.core.validators import MaxValueValidator, MinValueValidator
from datetime import datetime
from rest_framework import filters
```

```
def user_photos_dir(instance, filename):
    usnrme = f'{instance.phone}-{instance.username}'
```

Б қосымшасының жалғасы

```
folder_name = f"{usrnme}/{datetime.today().strftime('%d_%m_%Y')}/{filename}"
```

```
return folder_name
```

```
class institut(models.Model):
```

```
    name = models.CharField(max_length=100, blank=True, null=True)
```

```
    def __str__(self):
```

```
        return self.name
```

```
class кафедра(models.Model):
```

```
    name = models.CharField(max_length=100, blank=True, null=True)
```

```
    institut = models.ForeignKey(institut, on_delete=models.CASCADE,  
blank=True, null=True, related_name="cafedra")
```

```
    def __str__(self):
```

```
        return self.name
```

```
class User(AbstractUser):
```

```
    GENDER_MALE = 0
```

```
    GENDER_FEMALE = 1
```

```
    GENDER_CHOICES = (
```

```
(GENDER_MALE, _('Male')),
```

```
(GENDER_FEMALE, _('Female')),
```

```
)
```

```
    avatar = models.ImageField(upload_to=user_photos_dir, default="default.png",  
null=True, blank=True)
```

```
    email = models.EmailField(_('email address'), unique=True, blank=True,  
null=True)
```

Б қосымшасының жалғасы

```
birth_date = models.DateField(null=True, blank=True)
location = models.CharField(max_length=30, blank=True, null=True)
gender = models.SmallIntegerField(choices=GENDER_CHOICES,
                                   null=True, blank=True)
phone = models.CharField(max_length=30, blank=True, null=True)
rating_array = ArrayField(models.FloatField(null=True, blank=True,
                                             validators=[MinValueValidator(0.0),
                                                         MaxValueValidator(5.0)]),
                           null=True, blank=True, default=list,
                           )
work_place = models.CharField(max_length=30, blank=True, null=True)
cafedra = models.ForeignKey(cafedra, on_delete=models.CASCADE,
                             blank=True, null=True, related_name="cafedra")
job = models.CharField(max_length=30, blank=True, null=True)

def Allrating(self):
    sum_r = 0
    rat = 0
    for i in self.rating_array:
        sum_r += i
    rat = sum_r / len(self.rating_array)
    return rat

def comment(self):
    return self.userrat.all()

def university(self):
    return self.unver.all()

class university(models.Model):
    user = models.ForeignKey(User, on_delete=models.CASCADE, blank=True,
                             null=True, related_name="unver")
```

Б қосымшасының жалғасы

```
years = models.CharField(max_length=50, blank=True, null=True)
spec = models.CharField(max_length=100, blank=True, null=True)
name = models.CharField(max_length=100, blank=True, null=True)

def __str__(self):
    return self.user.username
```

С қосымшасы
(міндетті)
Бағдарламалық код үлгісі

Index.html – Мұғалімдерге көрінетін бастапқы бет

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,initial-scale=1.0">
    <link rel="icon" href="<%= BASE_URL %>satabayev_university_icon.ico">
    <title>Satbayev University</title>
    <style>
      body{
        margin: 0;
        padding: 0
      }
    </style>
  </head>
  <body>
    <noscript>
      <strong>We're sorry but Satbayev University doesn't work properly without
      JavaScript enabled. Please enable it to continue.</strong>
    </noscript>
    <div id="app"></div>
    <!-- built files will be auto injected -->
  </body>
</html>
```