

А.А. Демчишин, І.С. Купріянов

МЕТОД СИНТЕЗУ «ЖИВИХ» КАРТИН

Анотація. Під терміном «жива» картина в роботі розуміється зображення картини, яке, при перегляді через смартфон або інший цифровий пристрій з камерою, набуває елементів анімації, які в реальному світі відсутні.

В роботі розглянуто задачу розробки систематизованої сукупності кроків синтезу живої картини. Запропоновано метод, який включає: створення анімації, генерація маркеру, розміщення картини та маркеру на площині галереї; захоплення відео потоку; відстеження маркера доповненої реальності; позиціонування об'єктів доповненої реальності відносно маркера; відображення анімації на екрані пристрою.

Показано, що застосування наведеної послідовності алгоритмів комп'ютерного бачення дає можливість проводити стабільне розпізнавання маркеру, а методи проективної геометрії дозволяють віднайти розміри маркера та визначити площину його розташування для позиціонування анімації. Реалізація програмної системи синтезу у вигляді веб-застосунку дала можливість сформувати єдину базу програмного коду, який з успіхом виконується як на персональному комп'ютері, так і на мобільному пристрої.

Ключові слова: жива картина, доповнена реальність, розпізнавання маркеру AR, трекінг маркеру AR, кути Ейлера.

Постановка проблеми. В останні роки все більшого поширення набувають комп'ютерні технології доповненої реальності (Augmented Reality) [1,2]. Доповнена реальність використовується як інструмент для створення розважального контенту (маски/лінзи в Instagram, Facebook, Snapchat), в ігровій індустрії (елементи доповненої реальності з геолокацією в Pokemon Go, Ingress), B2B-рішеннях (інтерактивні каталоги магазинів IKEA, Lego), рекламі (реклама в стрічці новин Facebook, Apple).

Під «живою» картиною в даній роботі будемо розуміти зображення картини, яке, при перегляді через смартфон або інший цифровий пристрій з камерою, набуває елементів анімації, які в реальному світі відсутні. В галузі мистецтва доповнена реальність надає можливість органічно продовжити задум автора твору. За допомогою технологій AR, картини галереї отримують можливість бути наділеними додатковим змістом.

© Демчишин А.А., Купріянов І.С., 2020

Аналіз останніх досліджень. Синтез зображення доповненої реальності проводиться на основі поєднання реального зображення, яке отримано на основі цифрової зйомки, із зображенням віртуального об'єкту, розміри якого, ортоскопічно підлаштовуються під розміри інших об'єктів реального світу [3].

Позиціонування віртуального об'єкта відбувається на основі відстеження (трекінгу) маркера, який з ним асоціюється. Трекінг маркера проходить в режимі реального часу. На практиці [4] маркером виступає рамка у формі квадрата із вписаним в неї ідентифікатором об'єкту (рис.1). Для зменшення впливу варіації передачі кольору та інтенсивності освітлення на процес розпізнавання ідентифікатор подається у чорно-білому кольорі з контрастними та чіткими деталями. Мінімальні розміри таких деталей визначаються роздільною здатністю матриці камери та обчислювальною потужністю процесора, - більша роздільна здатність маркера потребує проведення більшої кількості обчислювальних операцій.

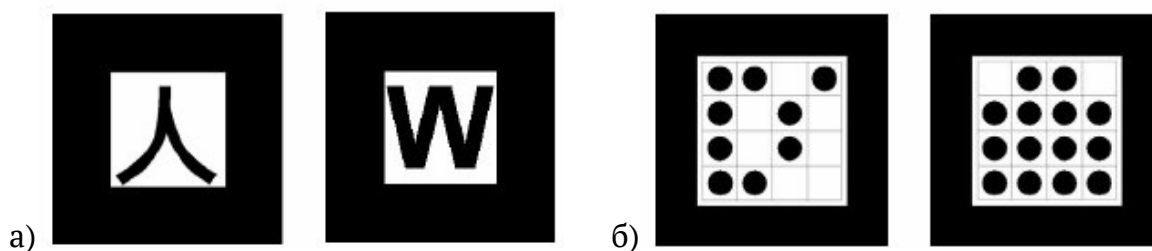


Рисунок 1 – Приклади квадратних маркерів: а) Siemens Corporate Research Marker System; б) ARToolKit (ATK) marker system

Формулювання цілей статті. Розробити систематизовану сукупність кроків синтезу живої картини, провести реалізацію кожного з кроків у вигляді алгоритму.

Виклад основного матеріалу дослідження. Підготовка до синтезу живої картини розпочинається зі створення мистецького твору у вигляді цифрової анімації, перший кадр якої збігається з картиною. Файл анімації розміщується безпосередньо на цифровому носії пристрою або у хмарі. Далі, до картини генерується індивідуальний маркер, який роздруковується, та розміщується на площині галереї поряд із картиною. Для перегляду анімації у доповненій реальності камера смартфона направляється на картину так, щоб маркер перебував в її полі зору.

Безпосередньо метод синтезу розпочинається із захоплення відео потоку, до кожного кадру якого застосовується послідовність алгоритмів комп'ютерного бачення з метою розпізнавання ідентифікатору та відстеження

маркера. На основі зафіксованих віконних координат кутів маркера проводиться зворотній розрахунок їх об'єктних координат. В даному випадку віконні і об'єктні координати розуміються в сенсі [5]. Наведений розрахунок буде виконуватись відповідно до швидкості зйомки камери: тобто, якщо камера налаштована на зйомку в режимі 30 кадрів на секунду, то час виконання алгоритму має бути меншим ніж 30мс. Відображення анімації у доповненій реальності проводиться у вигляді текстури, яка накладається на сторону прямокутника, виходячи з об'єктних координат маркеру.

Таким чином, метод синтезу живої картини можна подати у вигляді наступної послідовності кроків: підготовчий етап: створення анімації, генерація маркеру, розміщення картини та маркеру на площині галереї;

- 1) захоплення відео потоку;
- 2) відстеження маркера доповненої реальності;
- 3) позиціонування об'єктів доповненої реальності відносно маркера;
- 4) відображення анімації на екрані пристрою.

Якщо кроки 1 та 4 відносяться суто до програмування, то теоретична та алгоритмічна база кроків 2, 3 виглядає наступним чином.

Розглянемо алгоритм розпізнавання маркера AR. Алгоритм виходить із початкового зображення кадру, який отримано з відео потоку (рис. 2а). Візуалізацію маркеру проведено з використанням додаткового смартфона. Такий підхід значно полегшує проведення експериментів, адже прибирає необхідність роздруківки маркерів.

В контексті нівелювання впливу похибки передачі кольору, приведемо зображення кадру з кольорового до сірої палітри (рис. 2б). В якості варіанту реалізації змішування кольорів застосуємо рівняння (1).

$$E_Y = 0.299E_R + 0.587E_G + 0.114E_B \quad (1)$$

Коефіцієнти (0.299, 0.587, 0.114) відповідають так званому кольору білого С - кольору хмарного неба [6], за еталон якого обрано колір світіння абсолютно чорного тіла при температурі 6500 °С. Якщо ж демонстрація картини відбувається при освітленні з використанням ламп іншої температури, то рівняння (1) має бути відкоректовано.



Рисунок 2а – Вихідний кадр, що містить маркер (роздільна здатність кадру 1280x720)

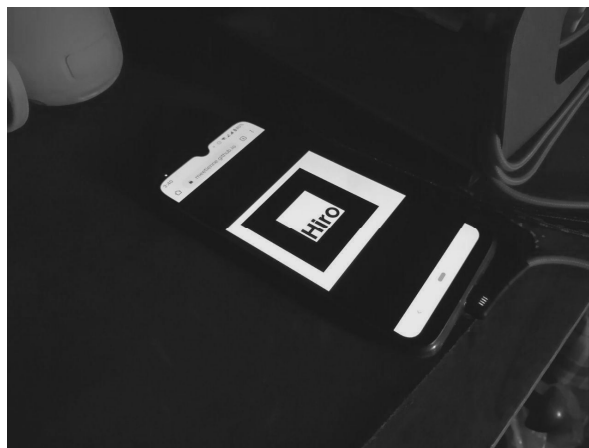


Рисунок 2б – Зображення у відтінках сірого

Для підвищення контрастності кадру до кожного пікселю зображення у 255 відтінках сірого застосуємо порогову функцію (2).

$$\theta(x) = \begin{cases} 0, & x < 128 \\ 255, & x \geq 128 \end{cases} \quad (2)$$

Як результат операції отримаємо чорно-біле зображення (рис. 2в).

Далі, проведемо виявлення зв'язних елементів на бінарізованому зображенні на основі алгоритму простої заливки [7]. Суть алгоритму полягає в тому, що білі пікселі та їхні сусіди заливаються «кольором» до тих пір, доки вся область зображення, обмежена границею з чорних пікселів, не буде залита. Замкнені області, які було виявлено на бінарізованому зображенні, показано на рис. 2г.

На наступному кроці виділимо контури ділянок зображення використовуючи алгоритм Собеля [8], та проведемо конвертацію контурів у полілінії (рис. 2д). Представлення контурів у вигляді поліліній надає можливість перейти до векторного представлення зображення.

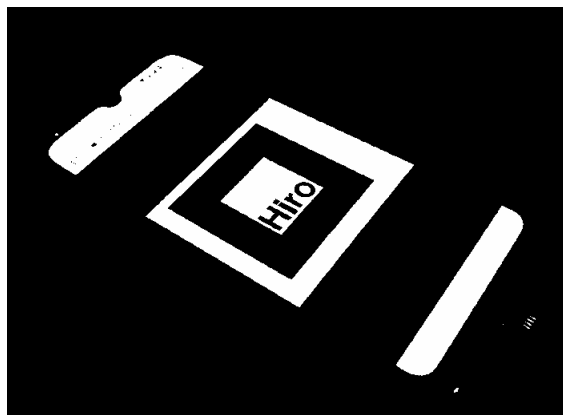


Рисунок 2в – Бінарізоване зображення

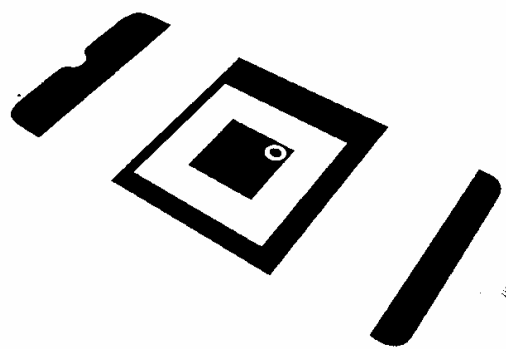


Рисунок 2г – Вигляд замкнених областей

Виділення кутів маркера на векторному зображенні було проведено ітеративним спрощенням замкнених поліліній до чотирикутника з координатами $P_A^W, P_B^W, P_C^W, P_D^W$ (алгоритм Дугласа-Пекера [9]). Приклад кадру з розпізнаними координатами чотирьох вершин подано на рис.2е.

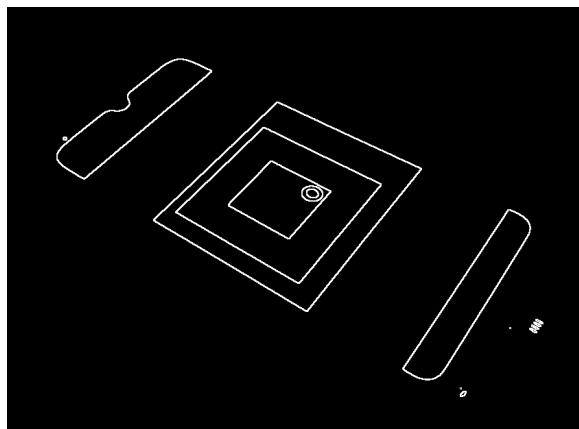


Рисунок 2д – Векторизоване зображення

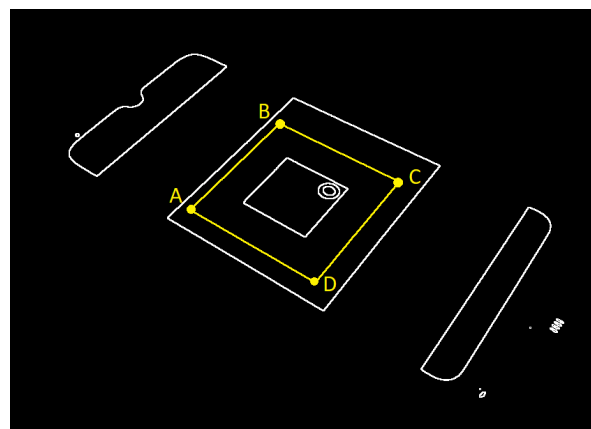


Рисунок 2е – Полілінія з чотирьох точок, яка відповідає маркеру доповненої реальності

Підсумовуючи, алгоритм розпізнавання маркера доповненої реальності записується наступним чином:

- 1) приведення зображення до палітри відтінків сірого;
- 2) бінарзація зображення (приведення зображення в чорно-біле);
- 3) виявлення замкнених областей;
- 4) виділення контурів;
- 5) виділення кутів контуру маркера;

Маючи віконні координати вершин чотирикутника $P_A^W, P_B^W, P_C^W, P_D^W$ на площині сенсора камери, алгоритм розрахунку їх об'єктних координат вигля-

дає наступним чином. Запишемо рівняння проекційних прямих, що проходять через вершини P_A, P_B, P_C, P_D та фокус камери з координатами $(0;0;0)$. Для цього задамося наступними параметрами камери смартфона: фокусна відстань лінзи f , кутове поле зору AOV , роздільна здатність сенсору $w \times h$, співвідношення сторін сенсору asp . Виходячи з цих параметрів, координати точок P_A, P_B, P_C, P_D дорівнюють:

$$\begin{cases} P_A = \left(\frac{{}^x P_A^w - w/2}{w/2} \cdot \text{tg}(AOV/2) \cdot f; \frac{{}^y P_A^w - h/2}{h/2} \cdot \frac{\text{tg}(AOV/2) \cdot f}{asp}; -f \right), \\ P_B = \left(\frac{{}^x P_B^w - w/2}{w/2} \cdot \text{tg}(AOV/2) \cdot f; \frac{{}^y P_B^w - h/2}{h/2} \cdot \frac{\text{tg}(AOV/2) \cdot f}{asp}; -f \right), \\ P_C = \left(\frac{{}^x P_C^w - w/2}{w/2} \cdot \text{tg}(AOV/2) \cdot f; \frac{{}^y P_C^w - h/2}{h/2} \cdot \frac{\text{tg}(AOV/2) \cdot f}{asp}; -f \right), \\ P_D = \left(\frac{{}^x P_D^w - w/2}{w/2} \cdot \text{tg}(AOV/2) \cdot f; \frac{{}^y P_D^w - h/2}{h/2} \cdot \frac{\text{tg}(AOV/2) \cdot f}{asp}; -f \right), \end{cases}$$

де індекси x, y у верхньому лівому куті вказують на компоненту вектору.

Загальні рівняння шуканих проекційних прямих виглядають наступним чином:

$$\begin{cases} A_A x + B_A y + C_A z = 0, & A_A = {}^y P_A^z P_A, B_A = {}^x P_A^z P_A, C_A = {}^x P_A^y P_A, \\ A_B x + B_B y + C_B z = 0, & A_B = {}^y P_B^z P_B, B_B = {}^x P_B^z P_B, C_B = {}^x P_B^y P_B, \\ A_C x + B_C y + C_C z = 0, & A_C = {}^y P_C^z P_C, B_C = {}^x P_C^z P_C, C_C = {}^x P_C^y P_C, \\ A_D x + B_D y + C_D z = 0, & A_D = {}^y P_D^z P_D, B_D = {}^x P_D^z P_D, C_D = {}^x P_D^y P_D, \end{cases} \quad (3)$$

де індекси x, y у верхньому лівому куті вказують на компоненту вектору.

Задамо еквівалент маркеру у вигляді квадратного шаблону з наступними однорідними координатами точок кутів: $N_A = (0;1;0;1)$, $N_B = (\sqrt{2}; \sqrt{2}; 0; 1)$, $N_C = (1;0;0;1)$, $N_D = (0;0;0;1)$. За допомогою паралельного переносу шаблону сумістимо кут N_D з точкою P_D (рис.3).

Будемо шукати таку матрицю повороту еквівалентного маркеру навколо точки P_D , яка дає найменше значення суми квадратів відстаней точок кутів шаблону від проекційних ліній (3):

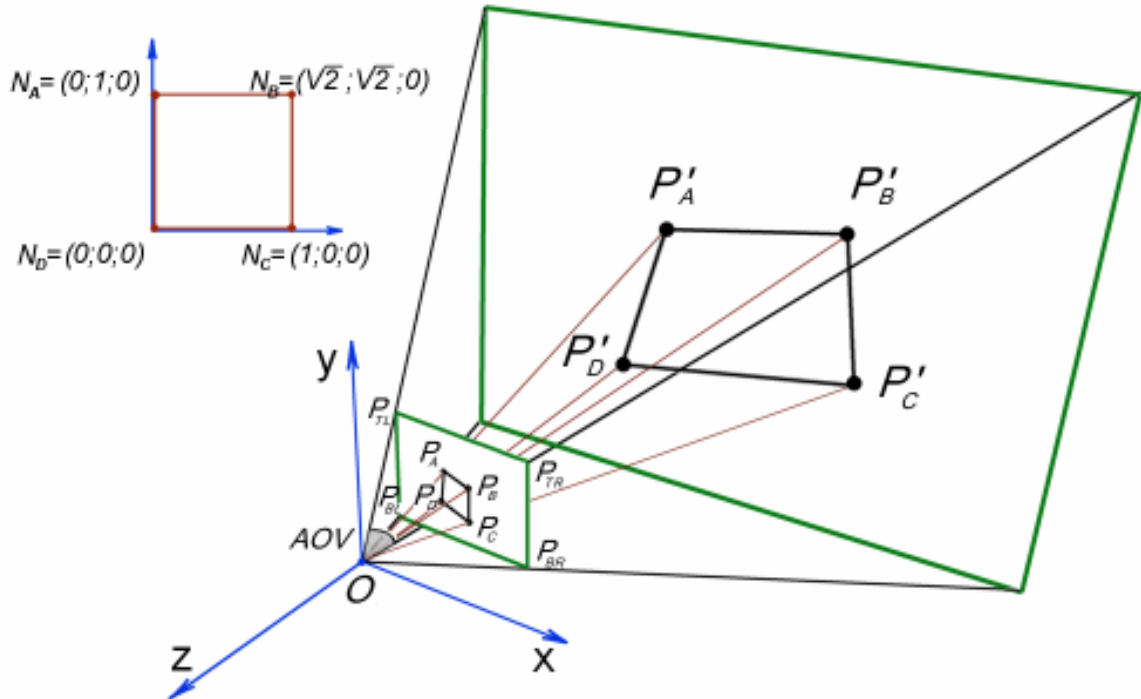


Рисунок 3 – Схема розрахунку координат маркеру

$$F(s, \alpha, \beta, \gamma) = \frac{((A_A; B_A; C_A), N'_A)^2}{A_A^2 + B_A^2 + C_A^2} + \frac{((A_B; B_B; C_B), N'_B)^2}{A_B^2 + B_B^2 + C_B^2} + \frac{((A_C; B_C; C_C), N'_C)^2}{A_C^2 + B_C^2 + C_C^2} \rightarrow \min,$$

$$N'_A = \mathbf{T} \cdot \mathbf{M} \cdot s \cdot N_A, N'_B = \mathbf{T} \cdot \mathbf{M} \cdot s \cdot N_B, N'_C = \mathbf{T} \cdot \mathbf{M} \cdot s \cdot N_C,$$

де s - коефіцієнт масштабування; \mathbf{M} - матриця повороту з головними стовпцями у вигляді, який подано нижче; \mathbf{T} - матриця переносу з головними стовпцями у вигляді (4); N'_A, N'_B, N'_C - радіус-вектори перетворених точок.

$$\mathbf{M} = \begin{bmatrix} \cos \alpha \cos \beta & \cos \alpha \sin \beta \sin \gamma - \sin \alpha \cos \gamma & \cos \alpha \sin \beta \cos \gamma + \sin \alpha \sin \gamma & 0 \\ \sin \alpha \cos \beta & \sin \alpha \sin \beta \sin \gamma + \cos \alpha \cos \gamma & \sin \alpha \sin \beta \cos \gamma - \cos \alpha \sin \gamma & 0 \\ -\sin \beta & \cos \beta \sin \gamma & \cos \beta \cos \gamma & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & {}^x P_D \\ 0 & 1 & 0 & {}^y P_D \\ 0 & 0 & 1 & {}^z P_D \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

Мінімізацію функції $F(s, \alpha, \beta, \gamma)$ проведено методом Флетчера-Рівза-Полака-Ріб'єра (FRPR) [10]. Початкові значення параметрів, що проходять оптимізацію, взято наступними:

$$s = h/2, \alpha = \beta = \gamma = 0.$$

При вказаних початкових значеннях шаблон займає половину висоти кадру, а матриця \mathbf{M} дорівнює одиничній, що відповідає ситуації, коли екран пристрою орієнтовано паралельно площині маркеру.

В загальному випадку технологія трекінгу на основі однієї камери не дає можливості розраховувати відстань від камери до маркеру. Оцінка відстані є можливою лише у разі, якщо розмір маркеру та параметри оптичної системи камери відомі точно. Метод синтезу живої картини не передбачає вимірювання дистанції до картини, тому z компонента задається виходячи з міркувань автора анімації та запланованої дистанції d огляду твору. Коефіцієнт масштабування s' , який визначає розмір фізичного маркеру може бути розраховано як:

$$s' = s \cdot d / f .$$

Знайдені кути Ейлера α, β, γ дають можливість розмістити тривимірну локальну систему координат, xOy площина якої співпадає з площиною, в якій лежить маркер (рис.4).

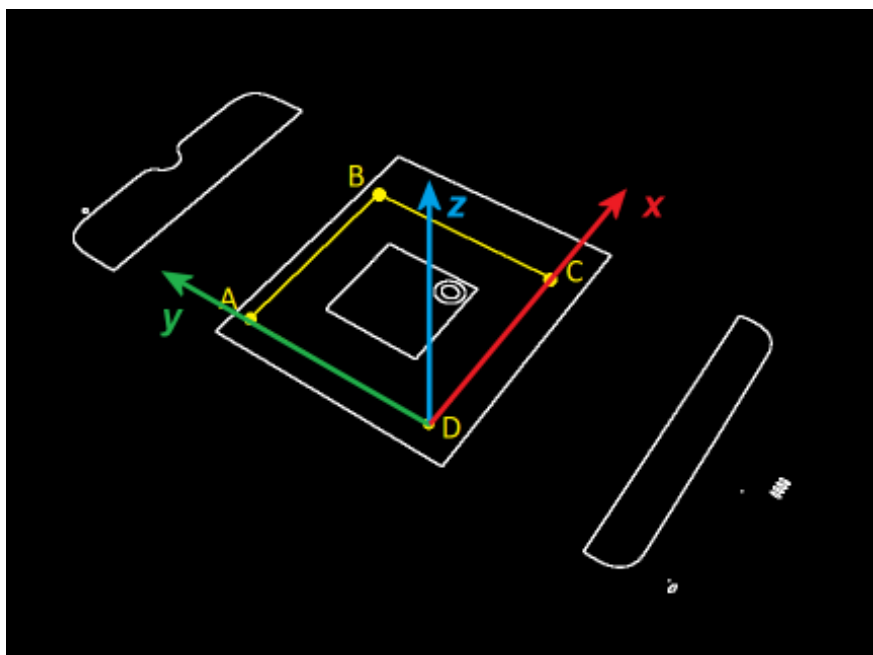


Рисунок 4 – Віднайдена координатна система маркера в об'єктних координатах

Програмна реалізація. В контексті даної роботи було реалізовано програмну систему синтезу живих картин у вигляді веб-застосунку з використанням мови програмування JS. На відміну від десктопного програмного забезпечення, яке необхідно розробляти для кожної операційної системи окремо, та яке по-

требує перекомпіляції (а в деяких випадках- транспіляції) для роботи на мобільних пристроях, реалізація системи у вигляді веб-застосунку дає можливість мати єдину базу програмного коду, який буде працювати у браузері більшості сучасних пристроїв.

Позиціонування та відображення об'єктів у тривимірній сцені проведено на основі фреймворка THREE.js. Результатом роботи застосунку є анімація, яка доповнює реальність, - точно перекриваючи зображення фізично існуючої картини (рис. 5).

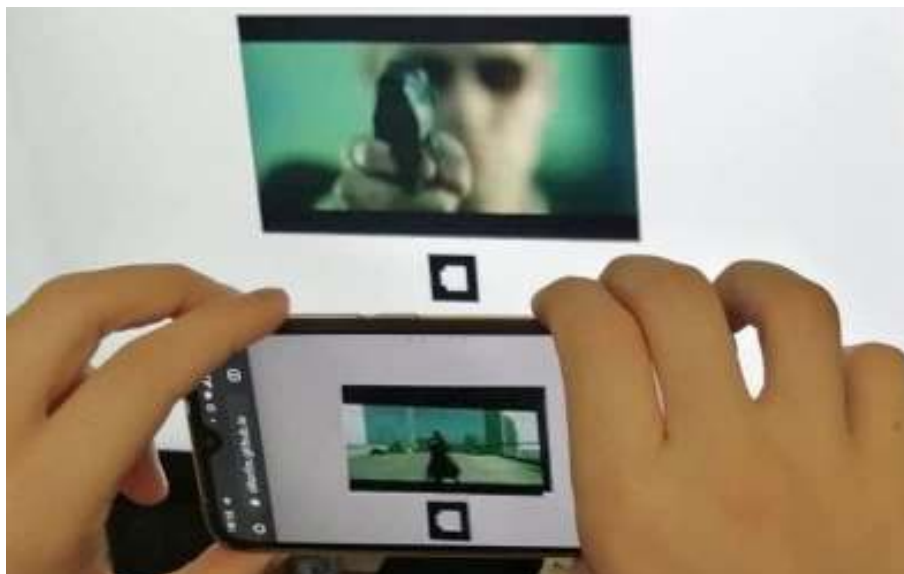


Рисунок 5 – Приклад синтезованого зображення (роздільна здатність камери пристрою 1280 x 720)

Висновки. В роботі розглянуто систематизовану сукупність кроків синтезу живої картини, яка включає: підготовчий етап: створення анімації, генерація маркера, розміщення картини та маркера на площині галереї; захоплення відео потоку; відстеження маркера доповненої реальності; позиціонування об'єктів доповненої реальності відносно маркера; відображення анімації на екрані пристрою.

Показано, що застосування наведеної послідовності алгоритмів комп'ютерного бачення дає можливість розпізнати маркер, а методи проективної геометрії дозволяють віднайти розміри маркера та визначити площину його розташування для позиціонування анімації.

Реалізація програмної систему синтезу живих картин у вигляді веб-застосунку надало можливість мати єдину базу вихідного коду програми, який з успіхом виконується як персональним комп'ютером, так і мобільним пристроєм.

ЛІТЕРАТУРА / LITERATURE

1. Kato H. Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System / H. Kato, M. Billinghurst. –Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality '99. –1999. – P. 85-94.
2. Безпалько О.С. Дослідження систем розпізнавання маркерів доповненої реальності / О.С. Безпалько. – System Research & Information Technologies. – 2020. – № 2(2020). – P. 55-66.
3. Milgram P., Kishino F. A taxonomy of mixed reality visual displays / P. Milgram, F. Kishino. – IEICE TRANSACTIONS on Information and Systems. –1994. Vol. 77 (12) – P. 1321-1329.
4. Siltanen S. Theory and applications of marker-based augmented reality / Siltanen S. – VTT Science 3. – Espoo 2012. – 198 p. + app. 43 p.
- Kessenich J., Sellers G., Shreiner D. OpenGL Programming Guide: The Official 5. Guide to Learning OpenGL, Version 4.5 / Kessenich J., Sellers G., Shreiner D. – Addison-Wesley Professional, 2016. – 976 p.
6. Седов С. А. Индивидуальные видео-средства / С. А. Седов. – К.: Наукова думка, 1990. – 752 с.
7. Гонсалес Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс — М: Техносфера, 2012 – 1104 с.
8. Вовк С.М. Методи обробки зображень та комп. зір : навч. посіб. / С.М. Вовк, В.В. Гнатушенко, М.В. Бондаренко. – Д. : ЛІРА, 2016. – 148 с.
9. Douglas D. and T. Peucker Algorithms for the reduction of the number of points required to represent a digitized line or its caricature / D. Douglas, T. Peucker. – Cartographica: The International Journal for Geographic Information and Geovisualization. – 1973. – Vol.10. – P.112-122.
10. Numerical Recipes in C: The Art of Scientific Computing / W. Press, S.Teukolsky, W. Vetterling, B. Flannery. – Cambridge univ. press. – 994 p.

REFERENCES

1. Kato H. Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System / H. Kato, M. Billinghurst. –Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality '99. –1999. – P. 85-94.
2. Bezpalko O.S. Doslidzhennia system rozpiznavannia markeriv dopovnenoii realnosti / O.S. Bezpalko. – System Research & Information Technologies. – 2020. – № 2(2020). – P. 55-66.
3. Milgram P., Kishino F. A taxonomy of mixed reality visual displays / P. Milgram, F. Kishino. – IEICE TRANSACTIONS on Information and Systems. –1994. Vol. 77 (12) – P. 1321-1329.
4. Siltanen S. Theory and applications of marker-based augmented reality / Siltanen S. – VTT Science 3. – Espoo 2012. – 198 p. + app. 43 p.

- Kessenich J., Sellers G., Shreiner D. OpenGL Programming Guide: The Official 5. Guide to Learning OpenGL, Version 4.5 / Kessenich J., Sellers G., Shreiner D. – Addison-Wesley Professional, 2016. – 976 p.
6. Sedov S.A. Individualnyie video-sredstva / S.A. Sedov. – K.: Naukova dumka, 1990. – 752 p.
7. Gonsales R. Tsifrovaya obrabotka izobrazheniy / R. Gonsales, R. Vuds — M: Tehnosfera, 2012 – 1104 p.
8. Vovk S.M. Metody obrobky zobrazhen ta kompiuternyi zir : navch. posib. / S.M. Vovk, V.V. Hnatushenko, M.V. Bondarenko. – D. : LIRA, 2016. – 148 p.
9. Douglas D. and T. Peucker Algorithms for the reduction of the number of points required to represent a digitized line or its caricature / D. Douglas, T. Peucker. – Cartographica: The International Journal for Geographic Information and Geovisualization. – 1973. – Vol.10. – P.112-122.
10. Numerical Recipes in C: The Art of Scientific Computing / W. Press, S.Teukolsky, W. Vetterling, B. Flannery. – Cambridge univ. press. – 994 p.

Received 06.12.2020.

Accepted 10.12.2020.

Метод синтеза «живых» картин

Под термином «живая» картина в работе понимается изображение картины, которое, при просмотре через смартфон или другое цифровое устройство с камерой, приобретает элементы анимации, которые в реальном мире отсутствуют.

В работе рассмотрено задачу разработки систематизированной совокупности шагов синтеза живой картины. Предложенный метод включает: создание анимации, генерация маркера, размещение картины и маркера на плоскости галереи; захват видео потока; отслеживание маркера дополненной реальности; позиционирование объектов дополненной реальности относительно маркера; отображение анимации на экране устройства.

Показано, что применение приведенной последовательности алгоритмов компьютерного зрения дает возможность распознать маркер, а методы проективной геометрии позволяют рассчитать размеры маркера и определить плоскость его расположения для позиционирования анимации.

Реализация программной системы синтеза живых картин в виде веб-приложения позволило сформировать единую базу программного кода программы, который с успехом выполняется как на персональном компьютере, так и на процессоре мобильного устройства.

Method of synthesis of "live" pictures

Augmented reality (AR) computer technologies have become widely spread in many areas of human life over the last years. The term "live" picture in the paper refers to an image of a real picture viewed through a smartphone or a camera of some other digital device. The image is augmented by virtual animation elements that are absent in the real world. In the field of art, augmented reality has become the tool for expanding an artist's idea in a natural way.

The paper describes a streamlined sequence of steps required for the synthesis of a live picture. The sequence includes the following steps: creation of a digital animation, generation of

an AR marker, staging the picture and the marker on a gallery's plane; capturing a video stream; AR marker tracking; positioning augmented reality objects relative to the marker; displaying the animation on a screen.

The authors show that application of the given sequence of computer vision algorithms makes it possible to recognize the marker reliably, while the usage of methods of projective geometry provides a way to determine the size of AR marker together with the coefficients of a plane holding the animation.

In general case, AR tracking technology that uses single camera has no yield of the distance from the camera to the marker. In the context of the method of synthesis of live pictures, z component of the marker is defined basing on the considerations of the animation author. Calculated Euler angles establish the position of a three-dimensional local coordinate system with xOy plane being coincident with the marker's plane.

The implementation of a software system for live pictures synthesis was made in the form of a web application. Web application paradigm made it possible to have a single base of a program code ready to be executed both on a personal computer and on a mobile device.

Демчишин Анатолій Анатолійович – доцент кафедри автоматизації проектування енергетичних процесів та систем, Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського".

Купріянов Іван Степанович – магістрант кафедри автоматизації проектування енергетичних процесів та систем, Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського".

Демчишин Анатолий Анатольевич – доцент кафедры автоматизации проектирования энергетических процессов и систем, Национальный технический университет Украины "Киевский политехнический институт имени Игоря Сикорского".

Куприянов Иван Степанович – магистрант кафедры автоматизации проектирования энергетических процессов и систем, Национальный технический университет Украины "Киевский политехнический институт имени Игоря Сикорского".

Demchyshyn Anatoliy – assistant professor, department of automation of design of energy processes and systems, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute".

Kupriianov Ivan – graduate student of the department of automation of design of energy processes and systems, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute".