

ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ

УДК 004.9

DOI: 10.34708/GSTOU.2022.14.13.001

ДЕСКРИПТИВНОЕ ПРЕДСТАВЛЕНИЕ 3D-ОБЪЕКТОВ ДЛЯ ИХ ИДЕНТИФИКАЦИИ В ПОДСИСТЕМЕ ИНФОРМАЦИОННОГО ПОИСКА АСУП

© А.Ю. Аликов, А.В. Калиниченко
СКГМИ (ГТУ), Владикавказ, Россия

Глобальные дескрипторы позволяют описывать форму всей поверхности 3D-модели и, как правило, используются для сравнения 3D-моделей друг с другом. В настоящей работе рассматривается возможность применения дескриптивного представления 3D-объектов для их идентификации в подсистеме информационного поиска в составе автоматизированной системы управления предприятием с целью «грубого» отсеивания заведомо отличающихся 3D-объектов, тем самым уменьшая объем входных данных, к которым в дальнейшем будут применены более точные и ресурсозатратные алгоритмы, что позволит увеличить скорость идентификации 3D-объектов без снижения точности.

Ключевые слова: 3D-объект, трехмерный объект, идентификация, дескриптор, АСУП.

С развитием технологий трехмерного сканирования, компьютерного зрения, аддитивных технологий в различных областях жизнедеятельности возникают задачи, связанные с идентификацией, классификацией 3D-объектов. Рассматриваемая в работе задача связана с одной из фундаментальных проблем компьютерного зрения – распознаванием трехмерных объектов. Некоторые репрезентативные задачи распознавания включают классификацию объектов, обнаружение и семантическую сегментацию. Данная задача имеет достаточно широкое практическое применение, в том числе связанное с обработкой заданных визуальных данных в области роботизированных производств, автономного вождения, систем виртуальной реальности.

Порой значимость информационных ресурсов сопоставима с основными ресурсами производства и оказывает существенное влияние на эффективность производства, что

подтверждает актуальность исследований, ориентированных на разработку алгоритмов, позволяющих повысить качество поиска информации, в том числе графической. Зачастую в условиях производства важно, чтобы используемые алгоритмы работали в режиме реального времени, что накладывает дополнительные требования на их временную сложность. Не менее актуальной остается и задача по разработке новых методов для решения задач классификации полученных моделей объектов. Эти методы должны характеризоваться построением алгоритмов идентификации 3D-объектов (сравнение с образцовыми моделями) с заданной точностью и соблюдением требований к быстродействию.

Информационные ресурсы автоматизированной системы управления предприятием (АСУП) включают в себя различную проектно-техническую документацию, положения, инструкции, пояснительные записки и про-

чее. Внедрение информационных технологий в сферу управления предприятием, необходимость анализа больших объемов данных при принятии решений, увеличение доли слабо-структурированной и неструктурированной информации, в том числе трехмерных моделей, приводят к необходимости повышения качества информационного поиска в автоматизированных поисковых системах [2]. Ключевые аспекты проектирования и разработки программных средств для создания анимационных роликов рассмотрены в работе [8]. В работе [2] предлагается математическая модель подсистемы информационного поиска в составе АСУП, основывающаяся на аппарате нечетких множеств и позволяющая формировать поисковый образ документа, более полно отражающий его содержание. Заметим, что в подсистеме информационного поиска поисковый запрос может быть представлен не только в текстовом виде, но и в виде документа-образца, например, трехмерной модели [3].

В настоящей работе рассматривается возможность применения дескриптивного представления 3D-объектов для их идентификации в подсистеме информационного поиска в составе АСУП с целью «грубого» отсеивания заведомо отличающихся 3D-объектов, тем самым уменьшая объем входных данных, к которым в дальнейшем будут применены более точные и ресурсозатратные алгоритмы, что позволит увеличить скорость идентификации 3D-объектов без снижения точности.

Под дескриптором понимается набор численных параметров (признаков), извлекаемых из объекта и позволяющих описать характеристики трехмерных объектов. Примерами дескрипторов могут являться форма 3D-объекта, его ориентация в пространстве и т.д. [6]. В дальнейшем при решении задачи идентификации вместо самих данных используются дескрипторы. К факторам, усложняющим процесс формирования глобальных дескрипторов, можно отнести различную ориентацию и масштаб 3D-объектов [6].

Обозначим основные требования, которые будем предъявлять к дескрипторам 3D-объектов:

- возможность уникальной идентификации, для чего различные объекты должны иметь различные дескрипторы;
- масштабируемость – зависимость размера дескриптора исключительно от способа его определения, а не от характеристик 3D-объекта, таких как цвет или форма;
- инвариантность к ориентации модели и масштабу;
- робастность к степени детализации модели;
- инвариантность к перенумерации вершин в сетке полигонов, из которой состоит 3D-объект.

Существуют различные подходы к построению дескрипторов. Так, использование дескрипторов, основанных на множестве двумерных силуэтов, предполагает выбор фиксированного числа направлений в пространстве, вдоль которых сохраняются контрастные снимки объекта. Далее для получения характеристик формы объектов на 2D-изображениях происходит анализ силуэтов (контуров). Контур, например, может аппроксимироваться ломаной, вершины какой-либо точки крайней кривизны или точки, которые добавляются для уточнения многоугольной аппроксимации. Для построения дескрипторов могут использоваться карты глубины с разных ракурсов. Представление 3D-объектов в виде воксельных моделей позволяет обобщить способы, которые использовались для формирования дескриптивного представления файлов растровой графики. Так, для описания прямоугольной структуры воксельных моделей могут быть использованы трехмерные матрицы, а значит и механизмы трехмерных дискретных сверток, и иные подходы, являющиеся обобщением подходов обработки двумерных растровых данных. Спиновые изображения позволяют кодировать распределение точек, которые находятся в окрестности некоторой опорной точки. Для построения спинового изображения требуются опорная точка на поверхности трехмерной модели, вектор нормали в этой точке, а также координаты всех точек, попавших в окрестность опорной точки [5]. Отметим также SIFT-дескриптор, использующий набор локальных признаков. Характеристики

дескриптора неизменны SIFT к масштабированию, перемещению и вращению изображения, а также частично инвариантны к изменениям освещения и аффинной или трехмерной проекции, признаки изображения эффективно выявляются с помощью поэтапной фильтрации. В ходе фильтрации идентифицируются стабильные точки, создаются ключи изображения, которые допускают локальные геометрические деформации. В дальнейшем ключи используются в качестве входных данных для метода индексации ближайшего соседа, который идентифицирует совпадения объектов-кандидатов. Окончательная проверка каждого совпадения достигается путем нахождения решения методом наименьших квадратов [1]. Дескриптор GLOH представляет собой модификацию SIFT-дескриптора. В дескрипторе GLOH фактически происходит вычисление SIFT-дескриптора, но используется полярная сетка разбиения окрестности на регионы, представляющие собой радиальные блоки. В результате получается вектор, содержащий

272 компоненты, который в дальнейшем для уменьшения размерности итогового вектора проецируется в пространство размерности 128 с помощью метода главных компонент (PCA).

Визуализация некоторых подходов к построению дескрипторов представлена на рис. 1. Очевидно, что каждый метод построения дескрипторов имеет свои преимущества и недостатки и подходит для конкретного контекста приложения. Не существует ни одного дескриптора, который лучше всего работает во всех ситуациях. Хотя некоторые обычно показывают более высокую производительность, чем другие в целом, для определенных случаев другие дескрипторы могут работать значительно лучше.

Немаловажным аспектом при анализе трехмерных графических данных является выбор формы их представления. Один и тот же объект может быть представлен с использованием различных способов представления трехмерных данных, причем эти представления могут кардинально отличаться, что ока-

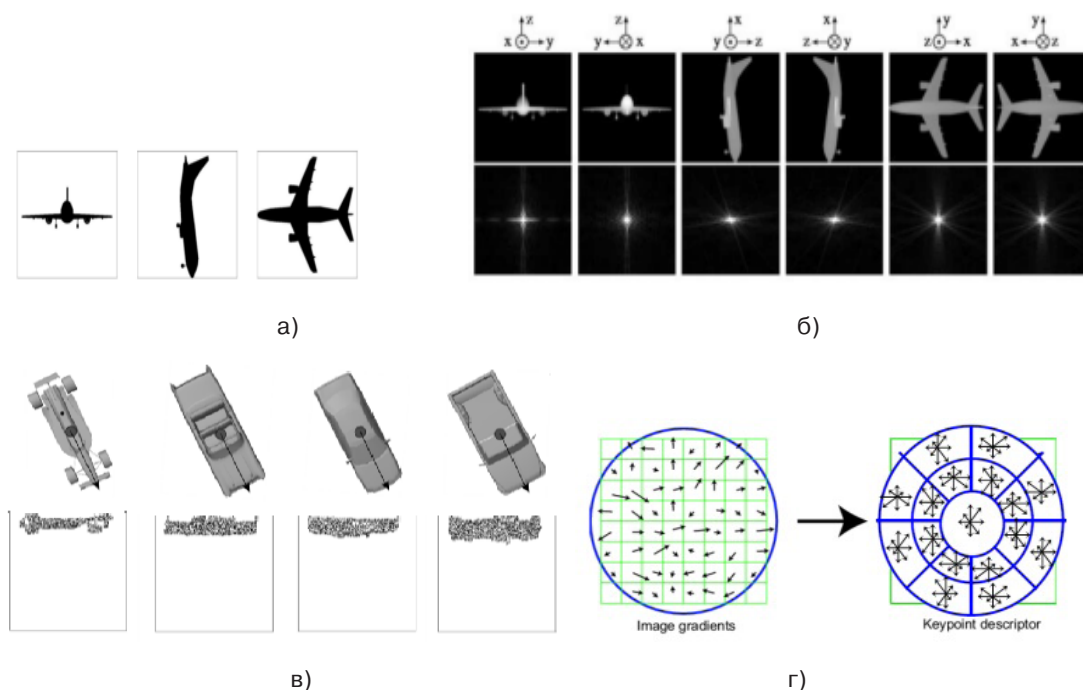


Рис. 1. Визуализация некоторых подходов к построению дескрипторов: а) набор контрастных силуэтов для дескриптора, основанного на множестве двумерных силуэтов, б) набор карт глубины 3D-модели (сверху) и их двумерного преобразования Фурье (снизу), в) нормализованные 3D-модели и графические представления их интегральных спиновых изображений, г) сетка разбиения окрестности на регионы в GLOH-дескрипторе

жет влияние на выбор модели обработки [5]. Достаточно распространены такие способы представления трехмерных данных, как полигональные модели, воксели, облака точек, модели, основанные на изображениях. Для хранения в памяти ЭВМ полигональных моделей могут использоваться способы, характерные для представления графов, например: множества вершин и множества ребер, соединяющих вершины. Достаточно часто используются вершины и тройки вершин, которые образуют элементарные полигоны. Обобщением пикселей на трехмерный случай являются воксели. Воксельная модель может быть представлена в виде трехмерного массива, в котором каждому элементу сопоставлен цвет и коэффициент прозрачности. Такой массив задает приближение трехмерного объекта с точностью, определяемой разрешением массива. Облако точек дает возможность представить трехмерный объект в виде неупорядоченного множества трехмерных радиусов-векторов. Для каждого вектора существует возможность указать цвет в качестве дополнительной составляющей. Подходы к моделированию и визуализации, основывающиеся на изображениях, генерируют трехмерную модель на основе набора двухмерных изображений.

В данной работе рассматривается задача построения дескрипторного представления для полигональных трёхмерных моделей. При таком представлении форма 3D-объектов в трехмерной компьютерной графике задается полигональной сеткой. Граними полигонов могут быть треугольники, четырёхугольники или многоугольники, причем каждая грань имеет нормаль, которая перпендикулярна поверхности грани. В качестве метода для составления дескрипторного представления будем использовать дескрипторы, основанные на распространении пучка лучей. Далее рассмотрим основные шаги алгоритма:

1. Прежде всего необходимо осуществить параллельный перенос компонентов 3D-объекта. Координаты вектора параллельного переноса вычисляются таким образом, чтобы центр массы 3D-объекта совпадал с началом координат.

2. На втором шаге выполняется масштабирование 3D-объекта. Коэффициент масштабирования выбирается таким образом, чтобы объект был вписан в сферу единичного радиуса, центр которой находится в начале координат.

3. Построение выпуклого правильного многогранника с центром тяжести в начале координат, описанного вокруг сферы.

4. Вычисление расстояния, которое проходит луч, исходящий из центра каждого полигона и направленный по нормали в сторону объекта, до столкновения с поверхностью 3D-объекта, нормаль которой направлена в сторону луча. Для лучшей фиксации геометрии из каждого полигона повторно строим луч, направленный из центра полигона по нормали в сторону 3D-объекта и определяем его расстояние до столкновения с поверхностью 3D-объекта, нормаль которой направлена в сторону, противоположную от луча.

5. Полученный набор признаков сортируется.

6. Наборы признаков принимают значения в пространстве признаков. Зададим в пространстве меру близости, для того чтобы стало возможным осуществление сравнения трехмерных объектов друг с другом, путем вычисления расстояния между соответствующими векторами признаков. Это дает возможность вычислять похожие друг на друга трехмерные объекты. Для определения расстояния будем использовать Евклидово расстояние:

$$p(x, x') = \sqrt{\sum_i^n (x_i - x'_i)^2}$$

7. Если объекты идентичны по форме (независимо от их размера и поворота), результат будет близок к единице.

Для реализации описанного алгоритма был использован Python API для среды 3D-моделирования Blender. Blender предоставляет встроенному интерпретатору модули Python, которые позволяют получить доступ к данным, классам и функциям Blender [9]. Python обращается к данным Blender так же, как к системе анимации и пользовательскому ин-

терфейсу. Это означает, что любой параметр, который можно изменить с помощью кнопки, также можно программно изменить средствами Python. Отметим некоторые модули Python API, использованные при реализации алгоритма. Доступ к данным из загруженного в Blender файла осуществляется с помощью модуля `bpy.data`. Модуль `bpy.data` позволяет получить доступ к объектам, сценам, материалам. Например:

```
>>> bpy.data.objects
<bpy_collection[3], BlendDataObjects>
>>> bpy.data.scenes
<bpy_collection[1], BlendDataScenes>
>>> bpy.data.materials
<bpy_collection[1], BlendDataMaterials>
```

Вызов операторов осуществлялся с помощью модуля `bpy.ops`. Под операторами понимаются инструменты, к которым пользователь обычно обращается с помощью кнопок, пунктов меню или сочетаний клавиш. С точки зрения пользователя это инструмент, но Python может осуществлять их запуск со своими настройками через модуль `bpy.ops`. Например:

```
>>> bpy.ops.mesh.flip_normals()
{'FINISHED'}
>>> bpy.ops.mesh.hide(unselected=False)
{'FINISHED'}
>>> bpy.ops.object.scale_apply()
{'FINISHED'}
```

Для доступа к данным, которые выбрал пользователь, применялся модуль `bpy.context`. Модуль `bpy.context` может использоваться для получения активного трехмерного объекта, сцены, настроек инструментов и многих других атрибутов, например:

```
>>> bpy.context.object
>>> bpy.context.selected_objects
>>> bpy.context.visible_bones
```

Модуль `bmesh` использовался для доступа к внутреннему API редактирования сетки Blender, включая данные о геометрии объекта, в том числе к нормальям и др. Отметим также, что Blender хранит множество наборов координатных данных для каждой части 3D-объекта. В большинстве случаев нас интересуют только два набора координат: глобальные координаты и локальные координаты, обозначим их G и L соответственно. Когда мы выполняем преоб-

разования над объектами, Blender сохраняет эти преобразования как часть матрицы преобразований T . Blender в какой-то момент применит матрицу преобразования к локальным координатам L . После того как Blender применит матрицу преобразования, локальные координаты будут равны глобальным координатам, а матрица преобразования будет идентификационной матрицей. В окне просмотра 3D мы всегда просматриваем глобальные координаты $G = T * L$. Мы можем контролировать момент, когда Blender применит преобразования, описанные выше, с помощью метода `bpy.ops.object.transform_apply()`. Мы использовали этот метод для того, чтобы легко выбирать определенные части объектов. Если мы отложим выполнение `bpy.ops.object.transform_apply()`, не запустив его и не выйдя из режима редактирования, то сможем сохранить два данных множества G и L . На практике глобальные координаты G очень полезны для позиционирования объектов относительно других, а используя локальные координаты L очень легко в цикле извлекать индексы. В коде, приведенном ниже, v – это список кортежей, который представляет нашу матрицу L , а `obj.matrix_world` – это матрица Python, которая представляет нашу матрицу преобразования T . Представляя данные `bpy.data.meshes[].vertices` в виде v , $v.co$ дает локальные координаты и `bpy.data.objects[].matrix_world * v.co` дает глобальные координаты.

```
def coords(objName, space='GLOBAL'):
    # Сохраним ссылку на объект bpy.data.
    objects
    obj = bpy.data.objects[objName]
    # Сохраним ссылку на объект bpy.data.
    objects[].meshes
    if obj.mode == 'EDIT':
        v = bmesh.from_edit_mesh(obj.data).verts
    elif obj.mode == 'OBJECT':
        v = obj.data.vertices
    if space == 'GLOBAL':
        # Возвращаем  $T * L$  как список кортежей
        return [(obj.matrix_world * v.co).to_tuple()
for v in v]
    elif space == 'LOCAL':
        # Возвращаем  $L$  как список кортежей
```



```

return [v.co.to_tuple() for v in v]

class sel:
    # Добавляем в класс ut.sel, чтобы использо-
    # вать в «object mode»
    def transform_apply():
        bpy.ops.object.transform_apply(
            location=True, rotation=True, scale=True)

```

Исследуем результаты алгоритма. Для этого проведём несколько сравнений различных 3D-моделей. Для моделей, являющихся вариациями одной (например, сравнение модели с её копией, повёрнутой в пространстве, или с моделями одного и того же объекта, но с различным количеством полигонов), результат близок к единице, для совершенно различных по форме моделей результат приближается к нулю.

Например, представленные на рис. 2 модели с различным количеством полигонов распознаны с коэффициентом схожести 0.97, т.е. не одинаковые, но достаточно похожие.

Увеличение количества полигонов в выпуклом правильном многограннике, очевидно, будет повышать точность вычислений, однако может значительно увеличивать время их вы-

полнения. Результаты эксперимента по оценке временных затрат на вычисление дескриптивного представления 3D-модели в зависимости от количества полигонов представлены на рис. 3. Зависимость имеет линейный характер.

В настоящей работе рассматривается подход к решению актуальной задачи идентификации 3D-объектов, представленных полигональными моделями. В результате проведенного исследования разработан программный модуль, осуществляющий построение дескрипторного описания 3D-моделей, а также позволяющий проводить «грубое» отсеивание заведомо отличающихся 3D-объектов, тем самым уменьшая объем входных данных, к которым в дальнейшем будут применены более точные и ресурсозатратные алгоритмы, что позволит увеличить скорость идентификации 3D-объектов без снижения точности. Заметим, что если предположить, что в качестве входных данных алгоритму подаются эскизы пользователя, например, один, два или три эскиза, которые аналогичны трем основным видам в инженерном чертеже, то предложенный подход может лечь в основу метода поиска трехмерной модели по ее двумерным эскизам.

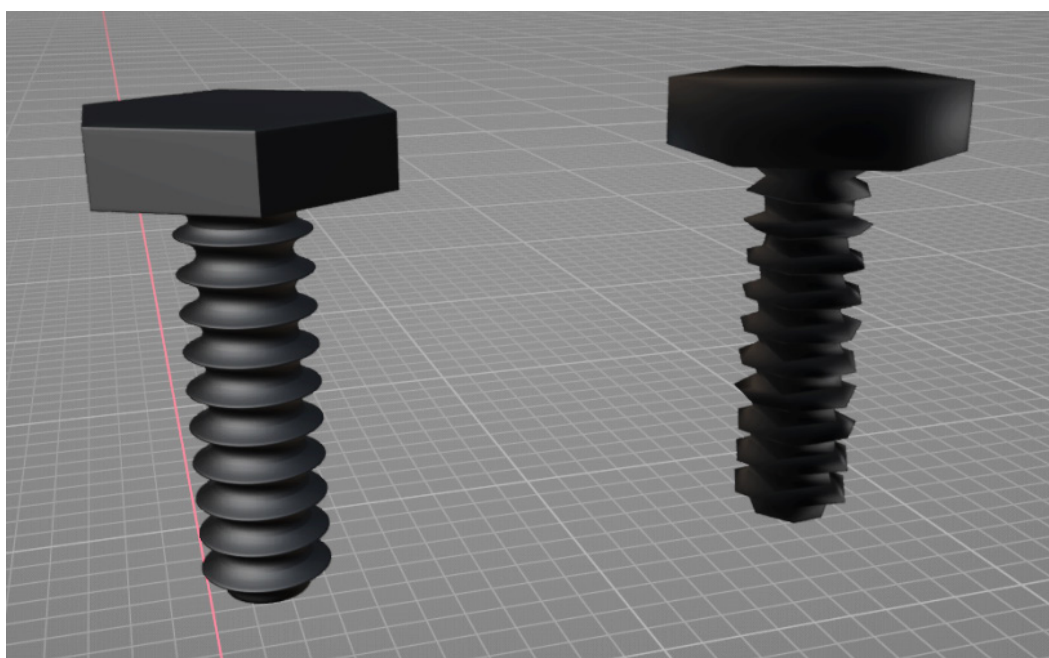


Рис. 2. Пример моделей с различным количеством полигонов

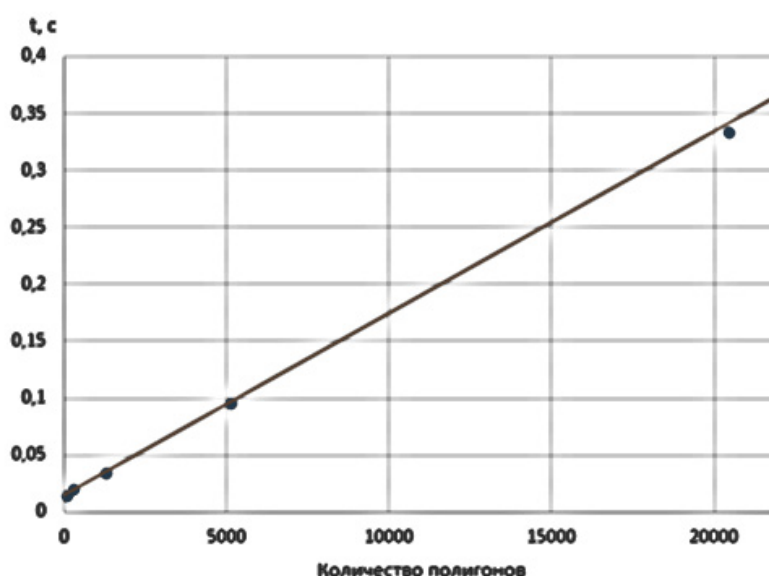


Рис. 3. График зависимости времени выполнения программы от количества полигонов в выпуклом правильном многограннике

ЛИТЕРАТУРА

1. David G. Lowe Object Recognition from Local Scale-Invariant Features // Proceedings of the Seventh IEEE International Conference on Computer Vision. V 2, pp. 1150-1157, doi: 10.1109/ICCV.1999.790410.
2. Аликов А.Ю., Калиниченко А.В. Математическая модель подсистемы информационного поиска в составе АСУП // Устойчивое развитие горных территорий. 2011. Т. 3. №4. С. 32-34.
3. Калиниченко А.В. О методах поиска по документу-образцу в коллекции электронных документов // ИТ-технологии: развитие и приложения. Материалы X Международной научно-практической конференции. 2009. С. 50-59.
4. Калиниченко А.В. О способах представления 3D-моделей для целей анализа данных // Тезисы докладов научно-практической конференции «Школа анализа данных». Владикавказ, 2021. С. 33-35.
5. Крыловецкий А.А., Черников И.С. Нормализация 3D-модели для вычисления интегрального спинного изображения // Известия Южного федерального университета. Технические науки. 2012. №6. С. 135-139.
6. Приступа А.В., Лапатин И.Л., Замятин А.В. Применение глобальных дескрипторов формы для идентификации 3D-объектов // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2019. №48. С. 57-66.
7. Сёмов А.А. Об одном подходе к распознаванию 3D-изображений // Труды международного симпозиума «Надежность и качество». 2013. Т. 1. С. 350-351.
8. Хасухаджиева Л.С.-А., Алисултанова Э.Д., Рутковский А.Л. Основопологающие аспекты разработки автоматизированной платформы для создания анимационных роликов // Вестник ГГНТУ. Технические науки. 2022. Т. 18. №2 (28). С. 24-32.
9. <https://docs.blender.org/api>

DESCRIPTION REPRESENTATION OF 3D OBJECTS FOR THEIR IDENTIFICATION IN THE INFORMATION SEARCH SUBSYSTEM OF ERP-SYSTEMS

© A.Yu. Alikov, A.V. Kalinichenko

NCIMM (STU), Vladikavkaz, Russia

Global descriptors allow you to describe the shape of the entire surface of a 3D model and are typically used to compare 3D models with each other. In this article, we consider the possibility of using a descriptive representation of 3D objects for their identification in the information retrieval subsystem as part of an automated control system for the purpose of "rough" sifting out obviously different 3D objects, thereby reducing the amount of input data, to which more accurate and resource-intensive algorithms will be applied in the future, which will increase the speed of identification of 3D objects without reducing accuracy.

Keywords: 3D object, identification, descriptor, ERP-systems.

REFERENCES

1. David G., Lowe 'Object Recognition from Local Scale-Invariant Features'. *Proceedings of the Seventh IEEE International Conference on Computer Vision*. V 2, pp. 1150-1157., doi: 10.1109/ICCV.1999.790410.
2. Alikov, A.Yu. and Kalinichenko, A.V. (2011) 'Matematicheskaya model' podsystemy informatsionnogo poiska v sostave ASUP'. *Ustoichivoe razvitie gornyykh territorii*. [Mathematical model of the information retrieval subsystem as part of the automated control system. Sustainable development of mountain territories]. V. 3, №4, pp. 32-34.
3. Kalinichenko, A.V. (2009) 'O metodakh poiska po dokumentu-obraztsu v kolleksii elektronnykh dokumentov'. *V sbornike: IT-tehnologii: razvitie i prilozheniya. Materialy X Mezhdunarodnoi nauchno-prakticheskoi konferentsii*. [On the methods of searching for a sample document in the collection of electronic documents. In the collection: IT technologies: development and applications. Materials of the X International Scientific and Practical Conference], pp. 50-59.
4. Kalinichenko, A.V. (2021) 'O sposobakh predstavleniya 3D modelei dlya tselei analiza dannykh.' *Tezisy dokladov nauchno-prakticheskoi konferentsii «Shkola analiza dannykh»*. [On the ways of representing 3D models for the purposes of data analysis. Abstracts of the scientific-practical conference "School of Data Analysis"], Vladikavkaz, pp. 33-35.
5. Krylovetsky, A.A. and Chernikov, I.S. (2012) 'Normalizatsiya 3D-modelei dlya vychisleniya integral'nogo spinovogo izobrazheniya'. *Izvestiya Yuzhnogo federal'nogo universiteta. Tekhnicheskie nauki*. [Normalization of a 3D model for calculating the integral spin image. Bulletin of the Southern Federal University. Technical science]. №6, pp. 135-139.
6. Semov, A.A. (2013) 'Ob odnom podkhode k raspoznavaniyu 3D izobrazhenii'. *Trudy mezhdunarodnogo simpoziuma «Nadezhnost' i kachestvo»*. [On one approach to 3D image recognition. Proceedings of the international symposium "Reliability and quality"]. Vol. 1, pp. 350-351.
7. Pristupa, A.V., Lapatin, I.L. and Zamyatin, A.V. (2019) 'Primenenie global'nykh deskriptorov formy dlya identifikatsii 3D ob'ektov'. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika*. [Application of global shape descriptors for identification of 3D objects. Bulletin of the Tomsk State University. Management, computer science and informatics]. №48, pp. 57-66.
8. Khasukhadzhieva, L.S.A., Alisultanova, E.D. and Rutkovsky, A.L. (2022) 'Osnovopolagayushchie aspekty razrabotki avtomatizirovannoi platformy dlya sozdaniya animatsionnykh rolikov'. *Vestnik GGNTU. Tekhnicheskie nauki* [Fundamental aspects of the development of an automated platform for creating animation videos. Herald of GSTOU. Technical sciences]. V. 18, №2 (28), pp. 24-32.
9. Available at: <https://docs.blender.org/api>