



АВТОМАТИЗАЦИЯ РАССТАНОВКИ ВИДЕОКАМЕР В ПОМЕЩЕНИИ

А.И. Лапо, Д.А. Мошкович, М.В. Дервис, А.А. Игнатович

Лицей Белорусского государственного университета, Беларусь

Аннотация. Анализируется автоматизация процесса расстановки видеокамер для распространения их суммарной видимости на наибольшую территорию на основе генетических алгоритмов. Работоспособность созданного программного обеспечения (ПО) проверена на примере реального помещения.

Ключевые слова. Расстановка видеокамер, генетические алгоритмы, программная реализация.

AUTOMATION OF PLACEMENT OF VIDEO CAMERAS IN THE ROOM

A.I. Lapo, D.A. Moshkovich, M.V. Dervis, A.A. Ignatovich

Lyceum of the Belarusian State University, Belarus

Abstract. Automation of the process of placing video cameras for the distribution of their total visibility to the largest territory on the basis of genetic algorithms is analyzed. The operability of the created software (SW) was tested on the example of a real room.

Keywords. Arrangement of video cameras, genetic algorithms, software implementation.

Введение

Согласно статистике, использование видеокамер на данный момент является одним из эффективных способов борьбы с преступностью. Кроме этого, использование систем видеонаблюдения позволяет организовать наблюдение за работой персонала внутри предприятия, а также обеспечить безопасную среду для сотрудников, которые работают с общественностью и могут стать жертвами разного рода преступлений.

В период эпидемии особо актуальным стало использование видеокамер для прокторинга. Прокторинг – это процедура контроля на онлайн-экзамене или тестировании, где за всем процессом наблюдает администратор – проктор. Он следит за действиями экзаменуемого с помощью веб-камеры и видит, что происходит на

мониторе его компьютера. Такая технология позволяет подтвердить личность кандидата, объективно оценить его знания, исключить различные уловки на экзамене. При этом испытуемый и проктор могут находиться в разных уголках планеты [1].

Проблема установки видеокамер для наблюдения за процессом стала актуальной и для Лицея БГУ, поскольку лицеисты участвовали в нескольких онлайн-олимпиадах с использованием прокторинга.

Однако вследствие увеличения числа используемых видеокамер возрастает количество проблем, связанных с их использованием. Наиболее очевидными из них являются:

- ограничение по количеству и качеству камер, связанное с конечностью материальных ресурсов предприятия или любого другого учреждения, нуждающегося в системе видеонаблюдения;
- сложность определения оптимальной расстановки заданного набора видеокамер, которая позволяет просматривать наибольшую территорию;
- затруднения, возникающие при попытке определить наилучшее расположение видеокамер для достаточно большого помещения.

Исходя из описанных выше проблем, возникает необходимость автоматизации размещения видеокамер. Поэтому можно утверждать, что данная тема является актуальной.

Гипотеза: автоматизация процесса расстановки видеокамер позволит распространить их суммарную видимость на как можно большую территорию.

Цель: разработать программное средство для автоматизации размещения камер в системе видеонаблюдения.

Задачи:

1. Изучить параметры видеокамер и качество их осмотра помещения.
2. Реализовать расстановку нескольких видеокамер с просчётом зон видимости и невидимости в помещении.
3. Найти и рассчитать параметры оптимизации размещения видеокамер.
4. Взяв за основу генетические алгоритмы [2], разработать алгоритм, оптимизирующий расстановку.
5. Составить программу, реализующую разработанный алгоритм.
6. Проверить и доказать работоспособность созданного программного обеспечения (ПО) на примере реального помещения.

Нами была рассмотрена программа JVSG. Однако предложенное ПО предназначено лишь для ручного проектирования систем видеонаблюдения и не

предоставляет возможности для автоматизации этого процесса. Иные рассмотренные альтернативы также предназначались для ручного проектирования, поэтому, в результате поиска, готовых решений, находящихся в публичном доступе, и полностью реализующих поставленные задачи, обнаружено не было.

Общее описание разработки

Разработанная программа включает в себя решение поставленных задач, а также удобный и интуитивно понятный графический пользовательский интерфейс [3-5].

Графический пользовательский интерфейс

Интерфейс программы (рис. 1) включает в себя:

- окно «Cameras», которое можно свободно перемещать, что позволяет адаптировать внешний вид программы под предпочтения пользователя;
- основную рабочую область, предоставляющую возможность быстро задавать очертания требуемой комнаты.

Окно «Cameras» предназначено для определения набора камер, которые в последующем будут использоваться для моделирования системы видеонаблюдения (см. приложение 1.1).

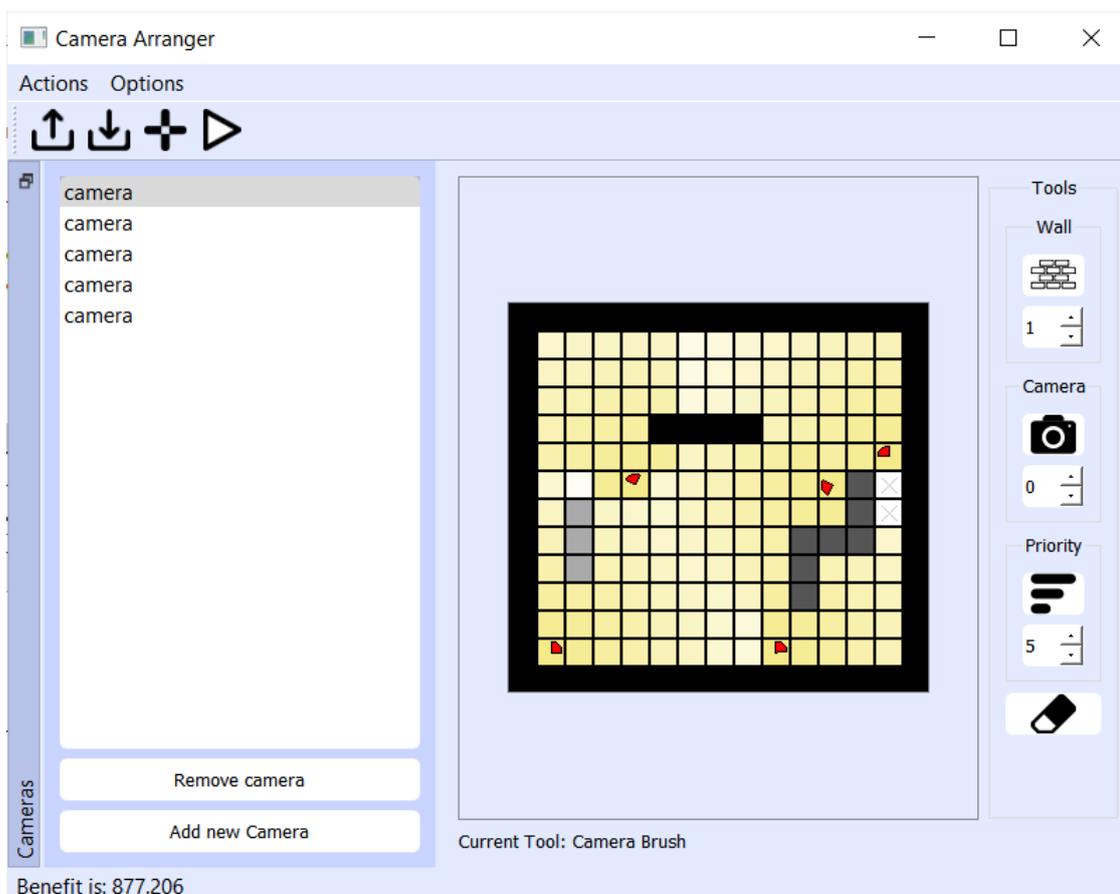


Рисунок 1 – Пользовательский интерфейс

Основная рабочая область, представляющая собой редактор помещения, содержит структурированное множество ячеек. В каждой ячейке может находиться камера, стена или иной загромождающий объект, установка которых осуществляется с помощью соответствующих инструментов (см. приложение 1.2):

- «Wall Brush» позволяет указать на то, что в некоторой части помещения находится стена или большой загромождающий объект определённой высоты. Предоставляет возможность создавать как одноклеточные стены, так и прямые линии.
- «Camera Brush» предоставляет возможность выбрать местоположение некоторой камеры из заданного набора. Также можно указывать для неё необходимый угол поворота.
- С помощью «Priority Brush» указывается важность осмотра той или иной части помещения.
- «Eraser of Walls» противоположен «Wall Brush», т. е. позволяет указывать на отсутствие стены в отмеченной ячейке. Также присутствует возможность удаления прямыми линиями.

Если в некоторой ячейке находится камера, то имеется возможность изменить угол её поворота.

В меню пользователю доступны пять действий:

- «New room» позволяет задать требуемые масштабы редактора, соответствующие размерам помещения без учета внешних стен. Также указывается высота, на которой размещаются камеры. (см. приложение 1.3).
- При выборе пункта «Run» (после инициализации комнаты и камер) начинает свою работу алгоритм расстановки камер.
- Меню «Options» содержит в себе пункт «Settings». Данный пункт позволяет задать количество популяций [2], которое влияет на точность и время работы алгоритма.
- Меню «Save as» и «Open room» дают возможность сохранять смоделированное помещение и найденную расстановку видеокамер.

Алгоритм

Изначально комната разбита на клетки, каждой из которой приписано значение ноль. В процессе расстановки камер клетка становится видимой. Степень видимости изменяет значение, стоящее в клетке. Сумма всех значений является основным

оценочным параметром расстановки камер – «выгода». Работа алгоритма направлена на максимизацию основного оценочного параметра.

Разобьем процесс генерации оптимальной перестановки на четыре основных этапа:

Формирование первого поколения представляет собой набор случайных расстановок видеокамер, предназначенный для дальнейшей обработки и совершенствования на каждом из последующих этапов. Для определенности можно считать каждую расстановку первого поколения константной.

Отбор подразумевает разделение очередного поколения на три части (группы) в зависимости от заранее рассчитанной суммарной выгоды, получаемой от всех расстановок группы.

Затем происходит замена группы наименее «выгодных» расстановок на те, суммарная выгода которых является максимальной. Оставшаяся группа расстановок подвергается мутации, которая представляет собой произвольное изменение некоторых параметров каждой из камер всех расстановок оставшейся группы.

Действия, производимые на каждом из перечисленных этапов, начиная со второго, повторяются заранее определённое количество раз для получения «хорошего» результата. Причём «хорошим» считается результат наиболее близкий к оптимальному.

На каждом этапе происходит перерасчёт параметра «выгода» в зависимости от степени освещённости клетки (см. приложение 2.1).

Также рассматривается возможность наличия больших загромождающих предметов в помещении. При рассмотрении взаимодействия камеры с большими объектами учитывается высота нахождения камеры, а также высота самого предмета.

С помощью определения у некоторых клеток меньшей «выгодности», чем у других, реализуется приоритетный осмотр помещения, основанный на стремлении просматривать те клетки, «награда» за просмотр которых больше.

Попытка улучшения работы алгоритма

В алгоритм добавлялась «встряска», представляющая собой удаление в определённой момент части особей текущего поколения и замену их на особи одного из предыдущих поколений. Однако сравнение работы различных вариантов программы показало, что такие действия лишь ухудшают работу алгоритма, из-за чего «встряска» была удалена.

Исследование разных типов алгоритмов и их параметров

Также для дополнительной оптимизации работы алгоритма и сравнения полученного нами варианта с его альтернативами мы исследовали некоторые зависимости.

Мы сравнили работу трех алгоритмов, а именно жадного, генетического и полного перебора [2]. Сравнение производилось по двум параметрам: времени работы и качеству получаемого результата (см. таблицу 1).

Таблица 1 – Результаты сравнения алгоритмов

	Жадный		Полный		Генетический					
Асимптотика	$O(10^5 * c * c * S)$		$O(10^3 * ((S * 120)^c))$		$O(p * c * 10^5)$					
	Время	Качество	Время	Качество	Время	Качество	n	m	c	p
Тест 1	0.03	83	0.03	83	0.15	83	5	5	1	100
Тест 2	1.37	315	inf	-	0.40	337	10	10	2	100
Тест 3	6.50	446	inf	-	0.65	454	10	10	5	100
Тест 4	55.49	1375	inf	-	4.79	1250	20	20	5	400
Тест 5	164.56	1883	inf	-	14.94	1550	20	20	10	800
Тест 6	344.58	2222	inf	-	29.84	1863	20	20	15	1200
Тест 7	inf	-	inf	-	99.65	4666	40	40	25	2000
Тест 8	inf	-	inf	-	115.43	5085	40	40	30	2000
Тест 9	inf	-	inf	-	120.03	5874	50	50	35	2000
Тест 10	inf	-	inf	-	122.34	12086	60	60	40	2000

В таблице n , m – высота и ширина комнаты, c – количество камер в заданном наборе, p – число популяций. В асимптотике S является площадью заданного помещения.

Анализ результатов, представленный в графической форме, приведен в приложении 3.1. Также была выявлена зависимость точности получаемых результатов от общего числа популяций. Наиболее подходящим количеством популяций является значение 600. При дальнейшем увеличении этого значения качество работы алгоритма изменяется незначительно (см. приложение 3.2).

Было проведено сравнение времени достижения определённого значения «выгодности» в зависимости от числа особей, составляющих каждое поколение. Исходя из полученных результатов, оптимальным количеством особей в поколении является 24. Результаты представлены в таблице 2, а также в приложении 3.3.

Таблица 2 – Количество особей

Количество особей	Время
8	13.4
16	23.87
24	10.16
32	23.65
40	16.18
48	28.47
56	17.51
64	34.77

Проверка работоспособности на реальном помещении

Для исследования работоспособности составленной программы было выбрано здание лица БГУ. Чтобы осуществить проверку в редакторе была создана модель одного из этажей на основе плана пожарной эвакуации (см. приложение 4.1). Далее мы сравнили вариант расстановки видеокамер, существующий в лицее на данный момент (см. приложение 4.2), с тем, который определил разработанный нами алгоритм (см. приложение 4.3). В результате сравнения, было установлено, что «выгодность» расстановки, полученной алгоритмом, выше существующей на 23% и составляет 4251.15. «Выгодность» существующий системы видеонаблюдения составила 3456.28.

Выводы

В результате исследования полученных результатов, можно утверждать, что гипотеза «автоматизация процесса расстановки видеокамер позволит распространить их суммарную видимость на как можно большую территорию» подтвердилась на рассмотренных нами примерах. Также использование генетических алгоритмов при моделировании систем видеонаблюдения позволяет оптимизировать процесс расстановки и сократить общее число расходов.

Список библиографических ссылок (на языке оригинала)

1. Нестеренко Е. Что такое прокторинг и чем полезен на экзаменах, тестировании и в обучении [Электронный ресурс]. Москва, 2021. URL: <https://finacademy.net/materials/article/proctoring> (дата обращения: 19.05.2021).
2. Генетический алгоритм. Просто о сложном [Электронный ресурс]. Москва, 2021. URL: <https://habr.com/ru/post/128704/> (дата обращения: 19.05.2021).
3. The future is written with Qt [Electronic resource]. Tokyo, 2021. URL: <https://doc.qt.io/> (date of access: 19.05.2021).

4. C++ language documentation [Electronic resource]. URL: <https://docs.microsoft.com/en-us/cpp/cpp/?view=msvc-160> (date of access: 19.05.2021).

5. The largest database of free icons available in PNG, SVG, EPS, PSD and BASE 64 formats [Electronic resource]. URL: www.flaticon.com (date of access: 19.05.2021).

References (на английском языке)

1. Nesterenko Ye. Chto takoye proktoring i chem polezen na ekzamenakh, testirova-nii i v obuchenii [What is proctoring and how is it useful for exams, testing and training] [Electronic resource]. Moscow, 2021. Available at: <https://finacademy.net/materials/article/proktoring> (date of access: 19.05.2021). (In Russian).

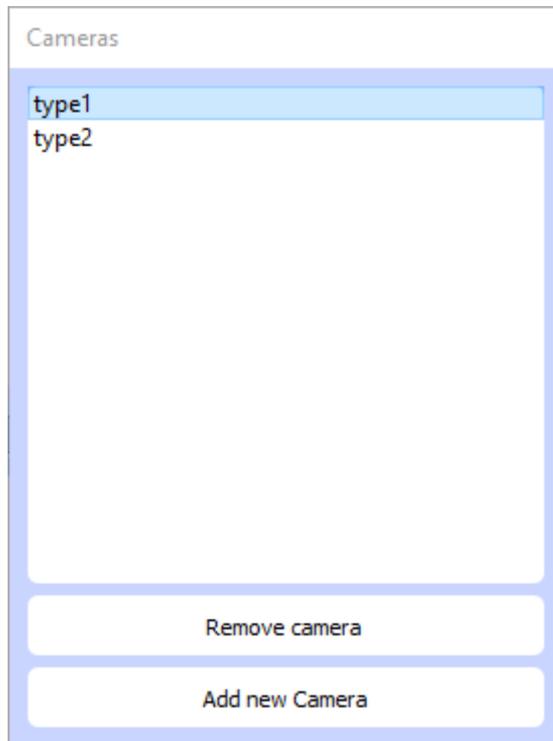
2. Geneticheskiy algoritm. Prosto o slozhnom [Genetic algorithm. Just about complicated] [Electronic resource]. Moscow, 2021. Available at: <https://habr.com/ru/post/128704/> (date of access: 19.05.2021). (In Russian).

3. The future is written with Qt [Electronic resource]. Tokyo, 2021. Available at: <https://doc.qt.io/> (date of access: 19.05.2021).

4. C++ language documentation [Electronic resource]. Available at: <https://docs.microsoft.com/en-us/cpp/cpp/?view=msvc-160> (date of access: 19.05.2021).

5. The largest database of free icons available in PNG, SVG, EPS, PSD and BASE 64 formats [Electronic resource]. Available at: www.flaticon.com (date of access: 19.05.2021).

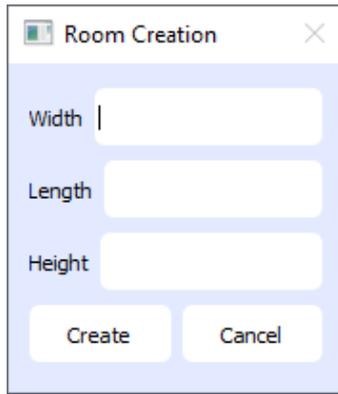
Приложение 1.1. Окно «Cameras»



Приложение 1.2. «Tools»



Приложение 1.3. «New Room»



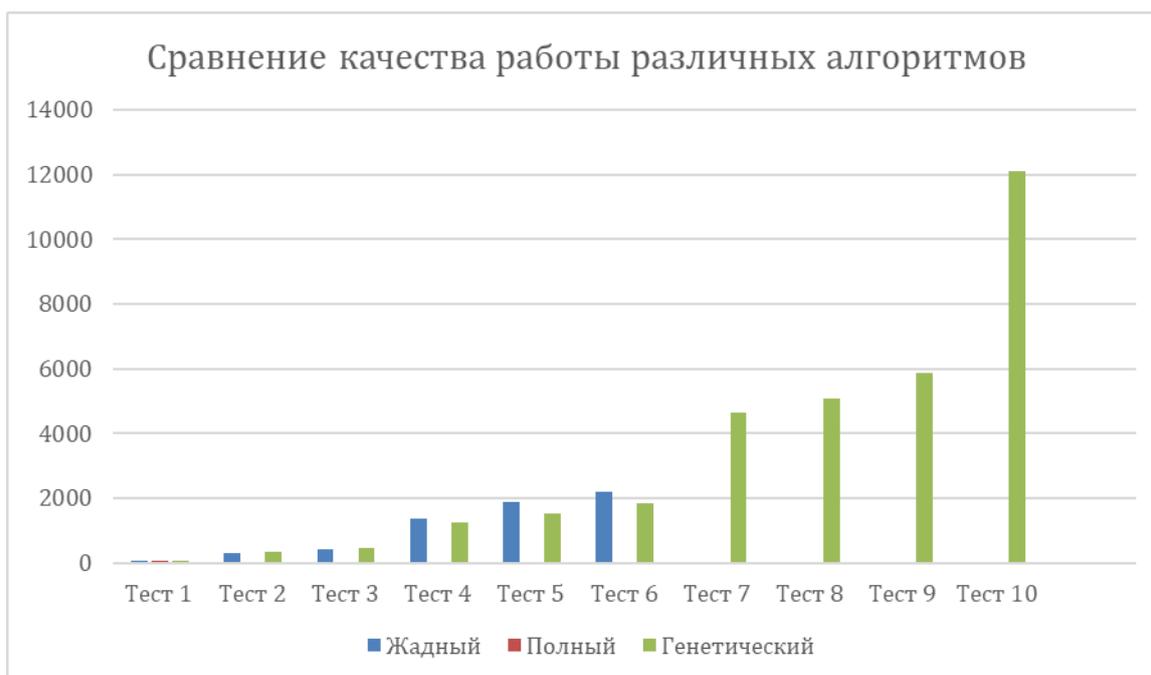
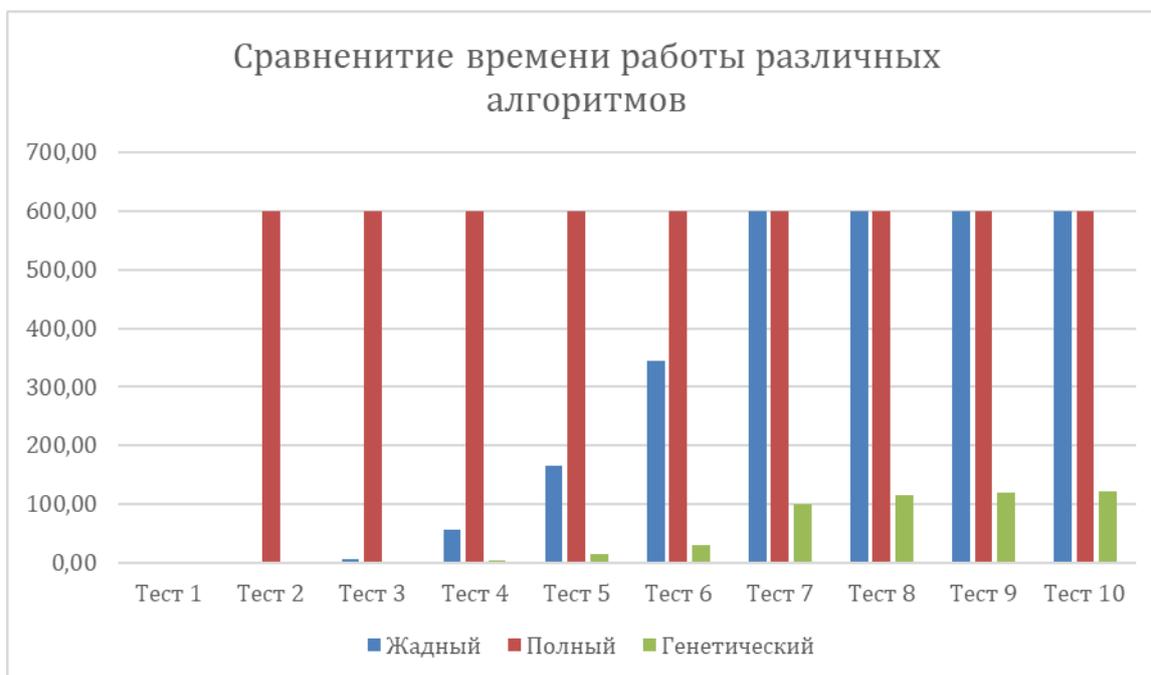
Приложение 2.1. Значения освещённости каждой клетки и параметр «выгода»

After:

#	#	#	#	#	#	#	#	#	#
#	2.00	1.90	#	5.00	4.88	4.53	4.00	#	#
#	3.00	2.88	#	6.00	5.84	5.39	4.76	#	#
#	4.00	3.84	#	7.00	6.76	6.17	#	#	#
#	5.00	4.76	#	8.00	7.59	6.76	#	#	#
#	6.00	#	#	9.00	5.00	4.90	4.61	4.17	#
#	7.00	6.00	5.00	4.00	6.00	5.88	5.53	5.00	#
#	0.00	0.00	#	0.00	7.00	6.84	6.39	5.76	#
#	#	#	#	#	8.00	7.76	7.17	6.39	#
#	0.00	0.00	0.00	#	9.00	8.59	7.76	6.84	#
#	0.00	0.00	0.00	#	10.00	9.00	8.00	7.00	#
#	#	#	#	#	#	#	#	#	#

Benefit 309.66

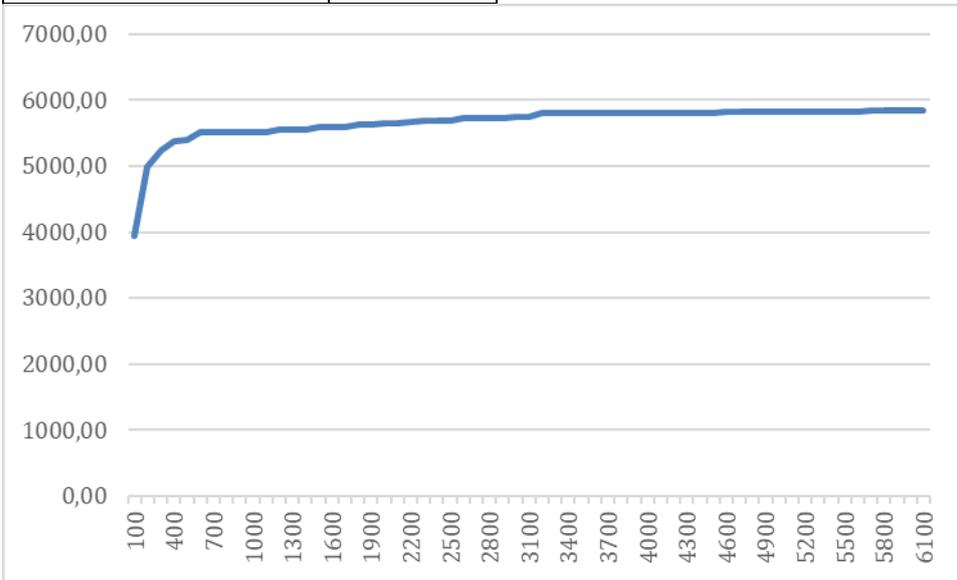
Приложение 3.1. Графическое представление результатов сравнения различных алгоритмов



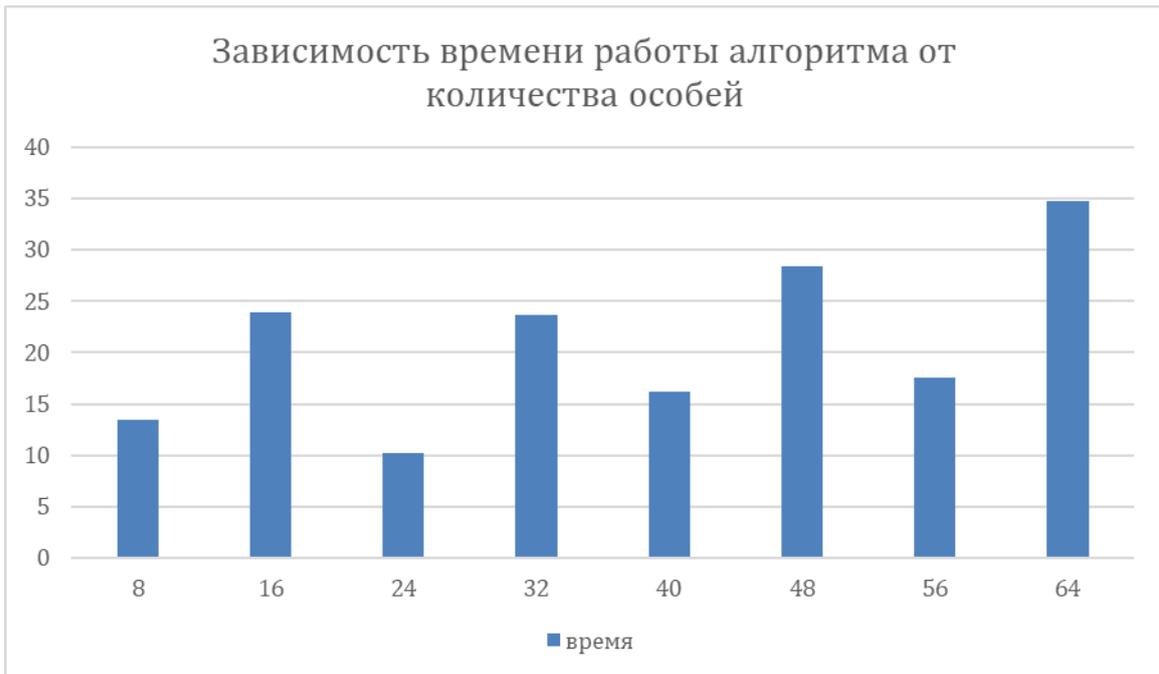
Приложение 3.2. Зависимость точности получаемых результатов от общего числа популяций

Количество популяций	Качество
100	3945.52
200	4988.55
300	5237.80
400	5374.70
500	5391.07
600	5510.40
700	5510.40
800	5510.40
900	5510.40
1000	5510.40
1100	5510.40
1200	5555.86
1300	5555.86
1400	5555.86
1500	5599.24
1600	5599.24
1700	5599.24
1800	5625.34
1900	5625.34
2000	5659.15
2100	5659.15
2200	5673.43
2300	5680.64
2400	5680.64
2500	5680.64
2600	5728.93
2700	5728.93
2800	5728.93
2900	5728.93
3000	5747.43
3100	5747.43
3200	5802.39
3300	5802.39
3400	5802.39
3500	5802.39
3600	5802.39
3700	5802.39
3800	5802.39
3900	5802.39
4000	5802.39
4100	5802.39
4200	5802.39
4300	5802.39

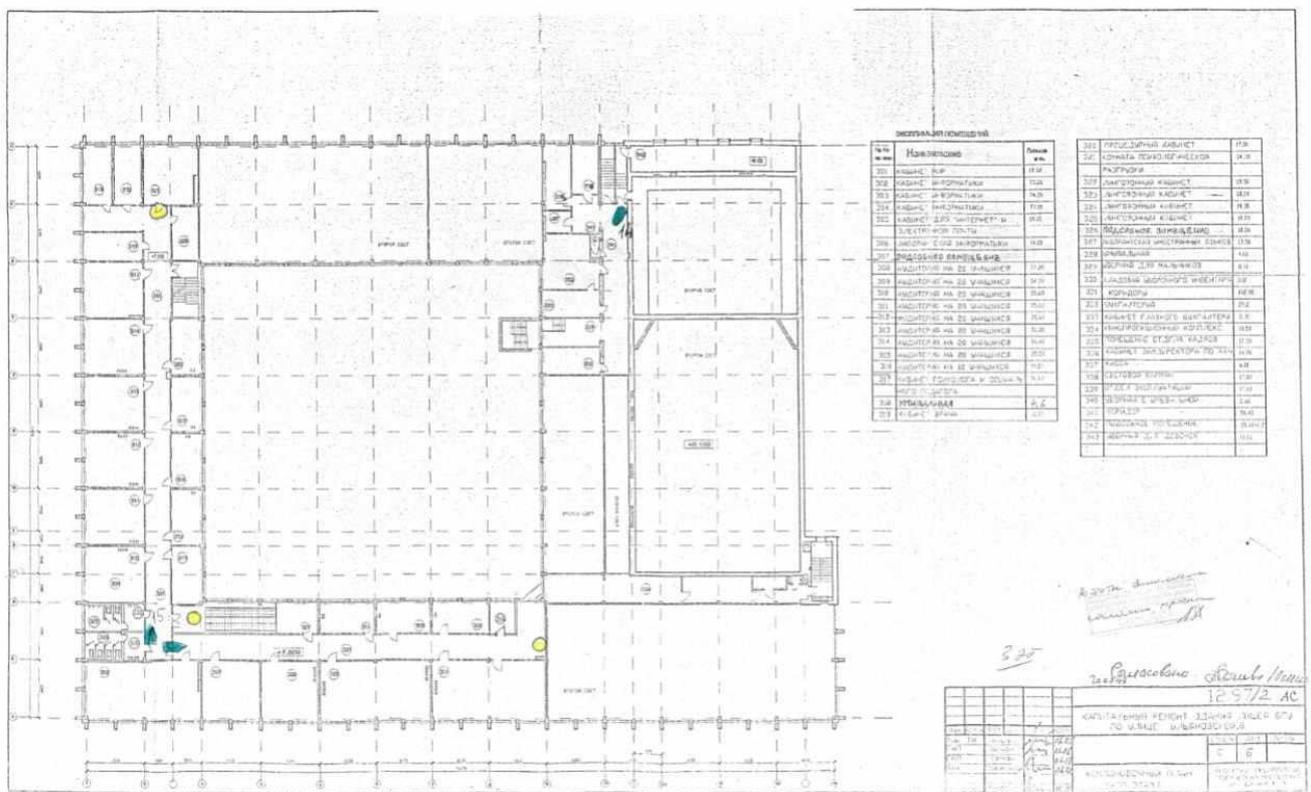
4400	5802.39
4500	5802.39
4600	5821.37
4700	5821.37
4800	5821.37
4900	5821.37
5000	5821.37
5100	5821.37
5200	5821.80
5300	5821.80
5400	5833.57
5500	5833.57
5600	5833.57
5700	5853.13
5800	5853.13
5900	5853.13
6000	5853.13
6100	5853.13



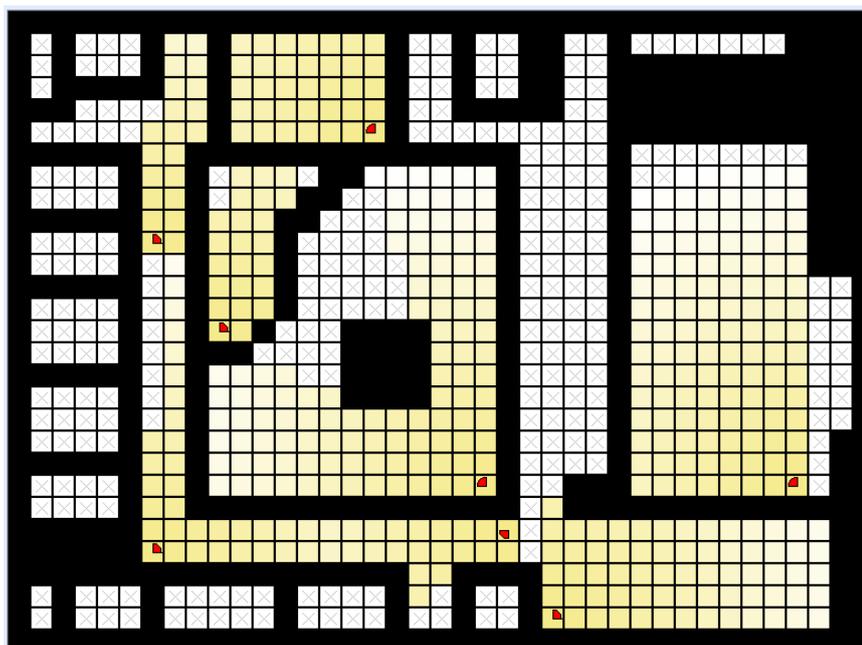
Приложение 3.3. Зависимость времени работы алгоритма от количества особей



Приложение 4.1. План системы видеонаблюдения Лицея БГУ (1 этаж)



Приложение 4.2. Существующая расстановка камер первого этажа здания Лицея БГУ



Приложение 4.3. Расстановка камер первого этажа здания Лицея БГУ, полученная в результате работы алгоритма

