

Д.А. Подопрхин

РЕАЛИЗАЦИЯ ИГРЫ «ЛАБИРИНТ» СРЕДСТВАМИ ВСІ

Введение

Значительное количество больных неврологическими заболеваниями не может полноценно контролировать движения определённых частей своего тела. Средства, создаваемые в рамках направления ВСІ (brain-computer interface), могут позволить им управлять и взаимодействовать со внешними устройствами, используя сигналы головного мозга.

Широкое распространение получили ВСІ-системы, базирующиеся на снятии электроэнцефалограмм (ЭЭГ). Благодаря появлению таких коммерческих устройств, как, например, Emotiv Eroc, данные системы вышли за пределы медицинских учреждений и стали доступны широкому кругу пользователей.

Существует несколько общих типов сигналов, которые используются для организации взаимодействия человека со внешними устройствами. Большинство из них перечислены в [3]. В данной работе в качестве управляющих сигналов используются сигналы, возникающие в ЭЭГ при моргании глазами. Несмотря на то, что моргание глаз в большинстве исследований рассматривается как артефакт, данный сигнал хорошо подходит для организации управления. В свою очередь, моргание глазами можно разделить на три типа: рефлексивное, добровольное (намеренное моргание) и спонтанное [1]. *Спонтанное моргание* связано с психо-физиологическим состоянием человека и в дальнейшем рассматриваться не будет. Намеренное моргание, в отличие от рефлексивного, затрагивает множество зон коры больших полушарий. В дальнейшем под морганием будем понимать *добровольное*, а *рефлексивное* относить к нейтральному состоянию.

При записи ЭЭГ наблюдаются скачки; пример приведён на рис. 1. Величина скачков и временной интервал, за который происходит скачок, зависит от длительности моргания и от того, насколько сильно были сжаты круговые мышцы глаза. Для классификации сигналов, связанных с морганиями, обычно используются методы, основанные на вычислении спектральных или статистических характеристик сигналов. Статистические методы считаются более эффективными в вычислительном плане, и поэтому лучше подходят для online-классификации¹.

¹Online-классификация — распознавание паттернов в реальном времени.

В ходе данного исследования была реализована игра, состоящая в проведении метки на экране монитора по простому лабиринту (*игра «Лабиринт»*). При разработке игры для распознавания морганий глаз был предложен метод формирования пространства признаков. Классификация сигналов осуществлялась хорошо зарекомендовавшими себя на практике методами SVM и Random Forest. Данные классификаторы будут описаны в разделе 2.3. Игра реализована в системе MATLAB.

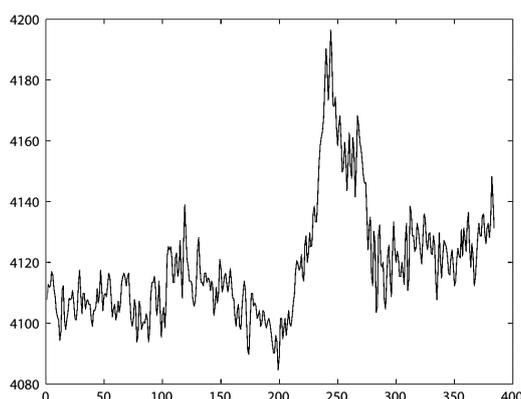
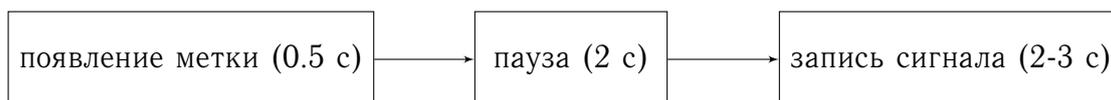


Рис. 1. Сигнал ЭЭГ, содержащий моргание (канал F7).

1 Описание эксперимента

Перед непосредственным этапом игры проводится обучение классификаторов. Данные для обучения представляют собой сигналы, содержащие распознаваемые классы: моргание правым и левым глазом, моргание двумя глазами, двукратное моргание двумя глазами, отсутствие моргания.

Испытуемому предлагалось в течение некоторого промежутка времени выполнять действия, изображённые на экране монитора (в дальнейшем данный промежуток времени будем называть *сессией*, а появление и выполнение каждого действия — *итерацией*). Каждая сессия включала в себя от 10 до 120 итераций, перерывы между итерациями составляли 2 с. Описание итерации представлено на диаграмме:



На каждой итерации испытуемому предлагалось моргнуть в соответствии с меткой (во избежание адаптации метки вспыхивали в случайном порядке согласно равномерному распределению). Типы меток: моргнуть левым глазом — «left wink», моргнуть правым глазом —

«right wink», моргнуть двумя глазами — «blink», моргнуть двумя глазами два раза — «double blink», не моргать — «neutral». Заметим, что класс «neutral» может содержать рефлексивные моргания. Испытуемый должен был моргать в тот момент, когда на экране появлялась надпись «запись сигнала». В некоторых сессиях присутствовала только часть меток.

Далее мы опишем графический пользовательский интерфейс, правила игры, а также модель управления.

Графический интерфейс представлен на рис. 2. Круг — цель, до которой необходимо добраться игроку, квадрат — игрок, а чёрные полосы — стены, через которые нельзя проходить. Квадрат движется с

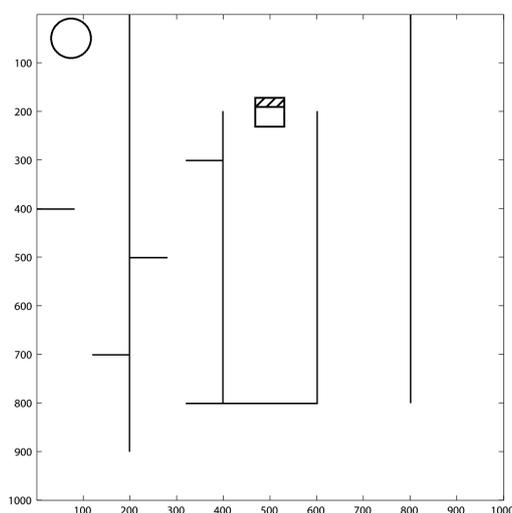
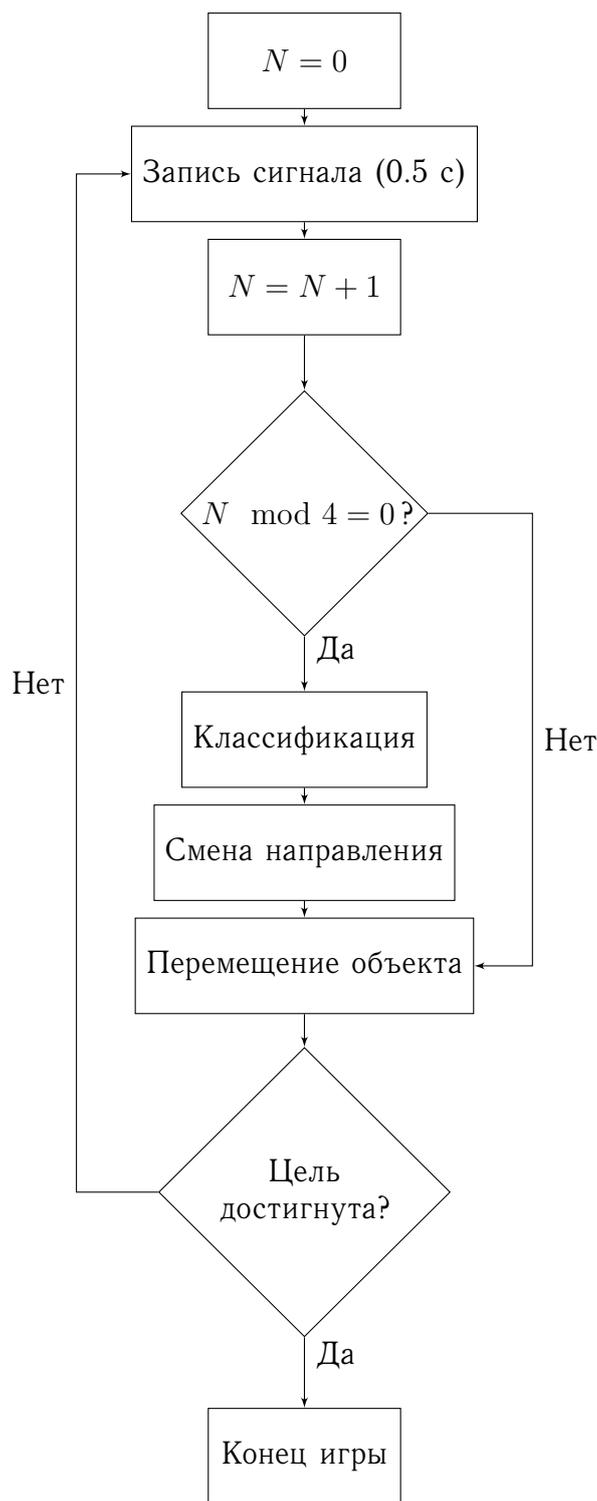


Рис. 2. Графический интерфейс игры «Лабиринт»

постоянной скоростью в выбранном направлении, а игрок может менять направление его движения. Доступны четыре направления: **вверх**, **влево**, **вниз** и **вправо** (угол между соседними направлениями составляет 90°). При старте задается направление **вверх**. Штрихованной линией указывается сторона квадрата, в направлении которой он движется. Чтобы сменить направление движения, т.е. совершить поворот заштрихованной стороны, необходимо моргнуть. При моргании левым глазом совершается поворот против часовой стрелки, правым — по часовой. На диаграмме представлен алгоритм работы программы:



Отметим, что смена направления происходит только в том случае, если классификатор обнаружил в сигнале моргание. В противном случае объект продолжает движение без смены направления. Классификатор направлений движения представляет собой композицию двух более простых классификаторов. Первый из них (classifier 1) определяет, содержит ли сигнал моргание, а второй (classifier 2) — каким

глазом оно выполнено.

Запись ЭЭГ данных осуществлялась с помощью 14-ти канального устройства Emotiv EEG Neuroheadset. Достоинствами данного устройства являются относительно низкая стоимость, доступность и простота использования. Недостатком является то, что устройство по качеству снимаемого сигнала уступает своим профессиональным аналогам.

Электроды устройства располагаются на местах $AF3$, $AF4$, $F7$, $F8$, $F3$, $F4$, $FC5$, $FC6$, $T7$, $T8$, $P7$, $P8$, $O1$ и $O2$ согласно международной системе размещения электродов «10-20». Частота дискретизации составляет 128 Гц.

2 Математическая задача

Пусть X — множество объектов, Y — конечное множество ответов, а $y : X \rightarrow Y$ — отображение, значение которого известно лишь на множестве объектов $\{x^1, \dots, x^l\} \subset X$. *Признаком объекта* называется отображение $f : X \rightarrow \mathbb{R}^n$, где \mathbb{R}^n — множество значений признака. Каждый объект x^i представлен *признаковым описанием* (f_1, \dots, f_k) . Совокупность $X^l = \{(x^i, y^i)\}_{i=1}^l$, где $y^i = y(x^i)$, называется *обучающей выборкой*. Необходимо построить отображение $a : X \rightarrow Y$, аппроксимирующее исходное отображение y на всём X . В данном исследовании объектами являются многомерные сигналы. Каждый многомерный сигнал состоит из 14-ти одномерных. Таким образом, за одну секунду аппаратура регистрирует $14 \times 128 = 1792$ признаков.

Задача формирования информативного признакового пространства меньшей размерности обусловлена:

- зависимостью признаков;
- наличием малоинформативных и шумовых признаков.

Данная задача является важной частью построения алгоритмов распознавания, поскольку от эффективности её решения существенно зависит решение всей задачи.

В дальнейшем каждый объект x будем описывать матрицей $x = \|f_{ij}\|_{14 \times n}$, где $f_i = (f_{i1}, \dots, f_{in})$ — сигнал, соответствующий i -му электроду; n — число отсчётов сигнала. Интервал между соседними отсчётами составляет примерно 7.8 мс.

2.1 Оценка качества классификации

Оценку полученных алгоритмов будем проводить следующим образом.

Разобьём выборку $X^l = \{(x^i, y^i)\}_{i=1}^l$ случайным образом на две равные части k раз (не ограничивая общности, полагаем, что l чётное) $X^l = X_{\text{train}}^i \sqcup X_{\text{test}}^i$, $i = \overline{1, k}$. Обучим классификатор $a(\cdot)$ на выборке X_{train}^i и оценим его качество на выборке X_{test}^i по формуле:

$$Q(X_{\text{train}}^i, X_{\text{test}}^i) = \frac{2}{l} \sum_{(x,y) \in X_{\text{test}}^i} [a(x) = y],$$

где $[x = y]$ — скобки Айверсона. Выполним данное действие для каждого разбиения и вычислим значение $Q(a)$:

$$Q(a) = \frac{1}{k} \sum_{i=1}^k Q(X_{\text{train}}^i, X_{\text{test}}^i),$$

которое и будет использоваться для оценки качества классификации.

2.2 Формирование признакового пространства

Предобработка сигнала включала в себя удаление постоянного амплитудного смещения (direct current offset) и фильтрацию. Согласно [2] спектральная плотность сигнала, содержащего моргания, сосредоточена в полосах 0.5-3 Гц и 8-13 Гц. Фильтрация производилась полосовым фильтром Баттерворта 5-го порядка с полосой пропускания $0.1 < f < 20$ Гц.

Признаковое пространство для classifier 1. На первом шаге необходимо отличить сигнал, содержащий моргание («all blinks»), от нейтрального состояния («neutral»). В данном разделе под морганием мы понимаем как моргание двумя глазами, так и моргание левым или правым глазом.

Перейдём к новому признаковому описанию следующим образом:

$$f_i^n = \max_{j=1, \dots, n-k} |f_{i,j+k} - f_{i,j}|.$$

Эксперименты показали, что наилучший результат достигается при рассмотрении двух каналов: второго (F7) и десятого (F8), и значения параметра k равном 10. Анализируя рис. 3, можно заметить, что данные два класса хорошо разделимы.

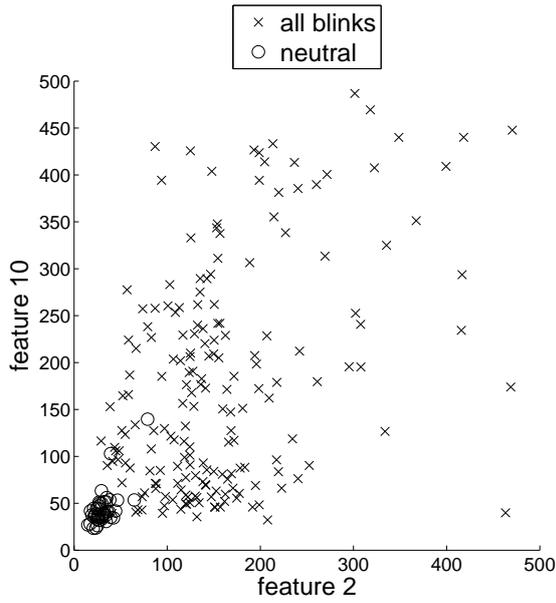


Рис. 3. Представление объектов классов «neutral» и «all blinks» в новом признаковом пространстве.

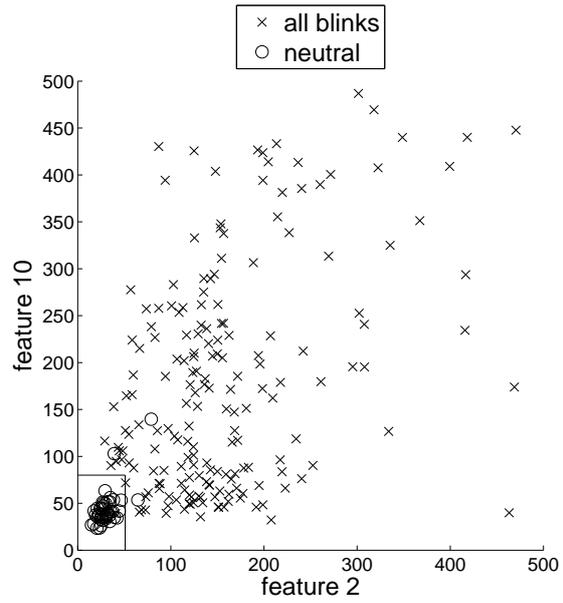


Рис. 4. Разделение признакового пространства на классы.

Признаковое пространство для classifier 2. В данном разделе будет предложен метод формирования признакового пространства и рассмотрен стандартный метод, основанный на FFT (Fast Fourier Transform), для классификации сигналов четырех классов «left wink», «right wink», «double blink» и «blink».

Method 1. Пусть $x = (x_1, \dots, x_n)$, $y = (y_1, \dots, y_n)$ — два одномерных сигнала. Тогда коэффициент корреляции между векторами вычисляется по формуле:

$$r(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}},$$

где

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i.$$

Далее X^l — исходная выборка. Для выявления закономерностей рассмотрим моргания, вызванные сильным сжатием круговых окологлазных мышц.

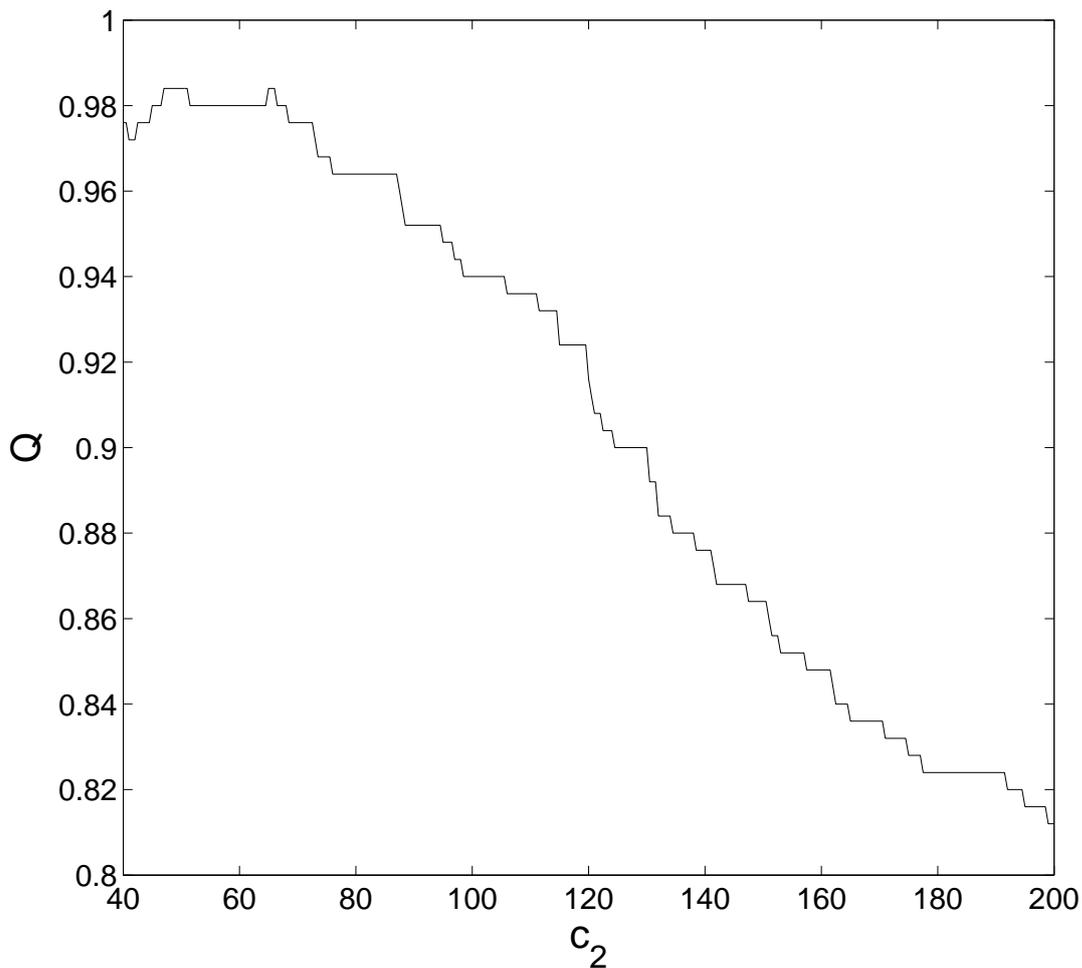


Рис. 5. Зависимость качества классификации от параметра c_2 .

Результат вычисления коэффициента корреляции между десятым и i -м каналом, где $i = \overline{1, 14}$, для классов «blink» и «left wink» представлен на рис. 6. Как можно заметить, сигналы неплохо различимы. В дальнейшем будем рассматривать моргания, соответствующие обычному сжатию круговых мышц.

Для каждой пары одномерных сигналов f_i, f_j вычислим r_{ij} , $1 \leq i < j \leq 14$. Это позволяет перейти к новому признаковому описанию, состоящему из 91 признака.

Пусть (f_{1j}, \dots, f_{kj}) — описание объекта x_j в новом признаковом пространстве. Произведём нормировку признаков:

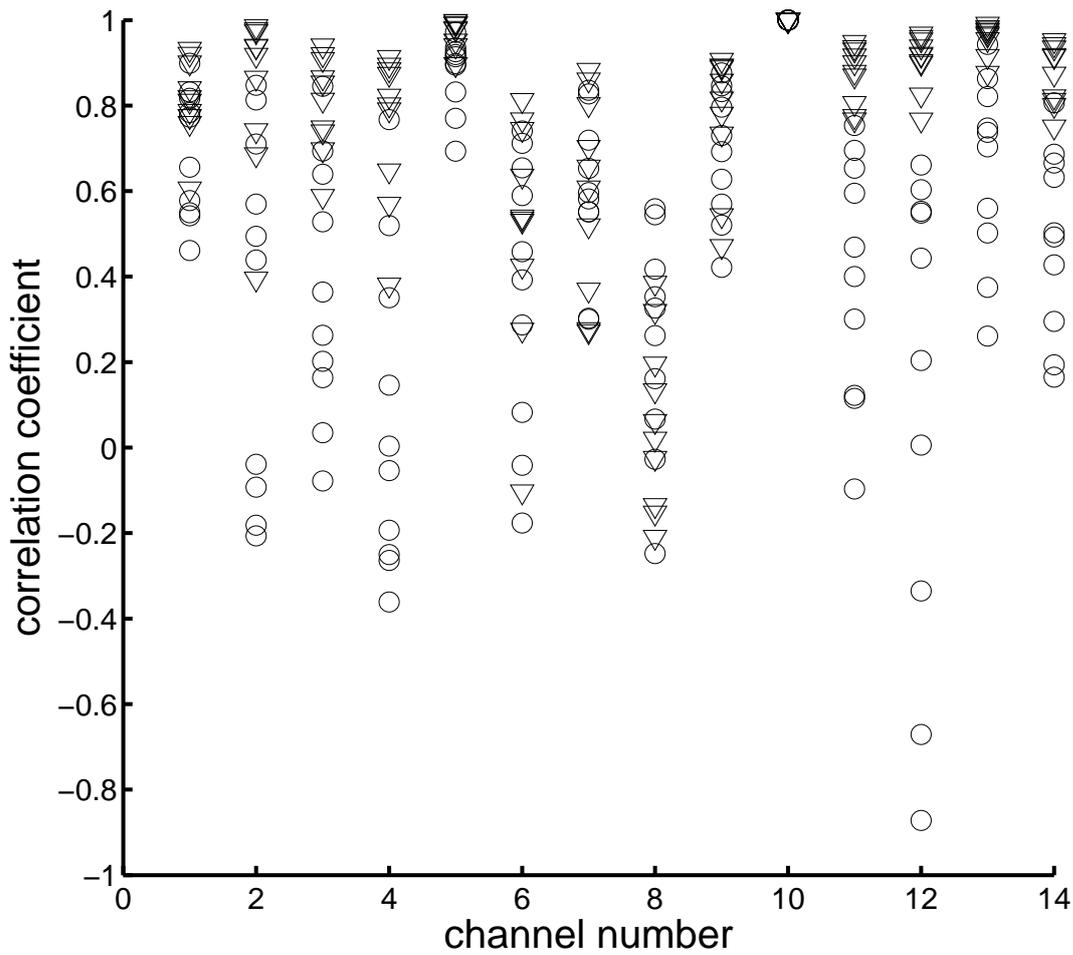


Рис. 6. Коэффициент корреляции между десятым и i -м каналом, где $i = \overline{1,14}$, для классов «blink» и «left wink».

$$f_{i,\max} = \max_{j \in \overline{1,l}} f_{ij}$$

$$f_{i,\min} = \min_{j \in \overline{1,l}} f_{ij}$$

$$f_{ij} = \frac{f_{ij} - f_{i,\min}}{f_{i,\max} - f_{i,\min}}$$

Method 2. Воспользовавшись FFT, вычислим его амплитудные характеристики. Поставим в соответствие каждому каналу f_i вектор (A_{i1}, \dots, A_{ik}) , где A_{ij} — амплитуда, соответствующая частоте j Гц. Исходя из экспериментов, параметр k примем равным 20. Тогда сигнал x описывается как вектор $(A_{1,1}, \dots, A_{1,k}, A_{2,1}, \dots, A_{14,k})$.

2.3 Алгоритмы классификации

После формирования признакового пространства происходит процедура классификации. Для классификации сигналов были использованы следующие алгоритмы: Random forest, SVM, RBF-SVM, которые кратко описаны ниже. Помимо указанных, были исследованы другие алгоритмы, такие как Neural Networks с различной архитектурой, AdaBoost и k-Nearest Neighbors [4], [10]. Однако данные методы в работе не представлены, так как они показали более низкий результат. Подбор всех приведённых параметров осуществлялся методом скользящего контроля.

Support Vector Machine. Основная идея SVM заключается в разделении данных $X \in \mathbb{R}^n$ на два класса путем нахождения вектора весов $\omega \in \mathbb{R}^n$ и порога b , максимизирующих минимальное расстояние между объектами выборки и гиперплоскостью, задаваемой уравнением: $y = \omega x + b$ [4]. Задача нахождения параметров сводится к оптимизационной задаче, которая имеет эффективное решение:

$$\begin{cases} \frac{1}{2} \langle w, w \rangle + C \sum_{i=1}^n \xi_i & \rightarrow \min_{w, w_0, \xi_i} ; \\ y_i (\langle w, x_i \rangle - b) \geq 1 - \xi_i, & i = 1, \dots, l ; \\ \xi_i \geq 0. \end{cases} \quad (1)$$

Переход к нелинейным разделяющим плоскостям осуществляется путём введения ядерных функций. Наибольшую популярность получили радиальные базисные функции:

$$f(x, y) = \exp(-R\|x - y\|).$$

SVM с данным ядром называется RBF-SVM. Реализация линейного SVM взята из библиотеки LibLinear [7], а RBF-SVM — из LibSVM [8].

Random Forest. Random forest — алгоритм машинного обучения, заключающийся в использовании комитета (ансамбля) решающих деревьев. В качестве ответа алгоритм выдаёт тот класс, за который проголосовало большинство деревьев [9]. Random forest сочетает в себе две основные идеи: метод бэггинга и метод случайных подпространств. Идея метода бэггинга заключается в том, что каждый алгоритм композиции строится по выборке, полученной из исходной методом бустреппинга. Бэггинг позволяет объединить несмещённые, но

чувствительные к составу обучающей выборки, алгоритмы в несмещённую композицию с низкой дисперсией. Метод случайных подпространств заключается в том, что при построении деревьев для разделения каждого узла на поддеревья используется только некоторый небольшой случайный набор признаков из множества всех признаков. Данный метод позволяет понизить корреляцию между деревьями. Реализация метода взята из системы MATLAB.

3 Результаты

Классификация сигналов «neutral» и «all blinks». В результате экспериментов была записана выборка из 339-ти объектов: 150 класса «neutral», 189 — класса «all blinks». Длительность большинства сигналов равна 2 с, однако встречаются сигналы и продолжительностью 3 с. Данные записывались в течение трёх дней. Основываясь на введённом выше признаковом описании, рассмотрим классификатор вида:

$$y = (f_2 \leq c_2) \& (f_{10} \leq c_{10}),$$

где c_2, c_{10} — пороговые значения для признаков f_2, f_{10} соответственно.

В результате экспериментов были получены оптимальные значения порогов $c_2 \in [47; 51]$, $c_{10} \in [60; 90]$, при которых качество классификации составляет 98.3% (рис. 4). Варьируя значения порогов, мы задаём чувствительность к слабым и рефлексивным морганиям. Чем больше значения порогов, тем менее чувствительный классификатор. Так как при управлении объектом нежелательно относить слабые или рефлексивные моргания, а также нейтральное состояние к классу «all blinks», то значительно завысим значения порогов. Анализируя рис. 4, нетрудно видеть, что наибольшее значение для качества классификации играет признак f_2 . Рассмотрим зависимость качества классификации от значения c_2 , которая представлена на рис. 5. Можно заметить, что при $c_2 \in [44; 76]$ качество классификации несильно отличается от наилучшего, и поэтому в дальнейшем положим $c_2 = 60$.

Классификация сигналов «left wink», «right wink», «blink» и «double blink». В результате экспериментов была записана выборка из 111-ти объектов: 26 — класса «left wink»; 32 — класса «right wink»; 22 — класса «double blink»; 31 — «blink». Данные были записаны в один день. Продолжительность сигналов — 2 с. Классификацию будем проводить с помощью SVM, RBF-SVM и Random Forest.

Заметим, что функционал Q не позволяет оценить ошибку классификатора на уровне отдельных классов, поэтому для решения данной проблемы рассмотрим матрицу ошибок (confusion matrix). Элемент матрицы, находящийся на пересечении j -ого столбца и i -ой строки, соответствует доле объектов класса j , отнесенных алгоритмом к классу i среди всех объектов выборки, выраженной в процентах. Матрицы ошибок для данных методов классификации представлены в таблицах 1 – 6. Для удобства в данных таблицах каждому названию класса поставлено в соответствие натуральное число: «left wink» – 1, «right wink» – 2, «double blink» – 3, «blink» – 4.

	1	2	3	4
1	22.5	0.6	0.6	1.0
2	0.6	27.3	0.7	1.9
3	0.1	0.5	10.1	5.3
4	0.0	0.0	8.1	20.6

Таблица 1. SVM + method 1.

	1	2	3	4
1	20.8	0.1	0.1	1.5
2	0.1	26.8	0.3	1.9
3	0.0	0.6	16.7	3.3
4	2.4	0.9	2.4	22.1

Таблица 2. SVM + method 2.

	1	2	3	4
1	22.5	0.6	0.0	0.3
2	0.7	27.0	0.5	1.5
3	0.0	0.4	10.2	7.2
4	0.1	0.4	8.8	19.8

Таблица 3. Random forest + method 1.

	1	2	3	4
1	19.4	0.7	0.3	0.3
2	1.5	26.4	0.7	1.5
3	0.1	0.4	15.5	2.9
4	2.2	0.9	2.9	24.2

Таблица 4. Random forest + method 2.

Наилучший результат показал SVM с method 2. Отметим, что при формировании признакового пространства с помощью method 1 классы «blink» и «double blink» плохо различимы между собой. Однако наибольший интерес в рамках работы представляет классификация на два класса: «left wink» и «right wink», т.к. именно с их помощью осуществляется управление.

Классификация сигналов «left wink» и «right wink». Данные записывались в течение двух дней, по три сессии в день (сессии различной продолжительности, т.к. испытуемый иногда уставал). В

	1	2	3	4
1	21.0	2.6	1.3	1.4
2	2.2	24.4	1.4	3.1
3	0.1	0.6	5.5	5.7
4	0.0	0.9	11.4	18.6

Таблица 5. RBF-SVM + method 1

	1	2	3	4
1	13.0	1.6	0.3	0.6
2	5.1	23.7	1.4	2.0
3	0.6	1.2	12.4	2.6
4	4.5	2.1	5.5	23.6

Таблица 6. RBF-SVM + method 2

результате было записано 80 объектов класса «left wink», 79 — класса «right wink». Длительность сигналов составляла 2 с. Для классификации рассмотрим алгоритмы Random Forest, SVM, RBF-SVM. Наилучшее качество для различных классификаторов представлено в таблице:

Алгоритм решения	Q
RBF-SVM ($R = 0.5$) + method 1	0.93
Random forest (600 деревьев) + method 1	0.97
SVM + method 1	0.95
SVM + method 2	0.90
RBF-SVM ($R = 0.125$) + method 2	0.83
Random forest (600 деревьев) + method 2	0.93

Отметим, что при классификации на два класса, «left wink» и «right wink», наилучший результат показал алгоритм Random Forest с предложенным методом формирования признакового пространства (method 1). Качество классификации достигает 97%.

4 Выводы

Для достижения наилучшего результата желательно каждый раз, когда электроэнцефалограф заново надевается на голову, переобучать классификатор на непродолжительной сессии из 20 объектов. Предложенный метод при классификации на 4 класса показывает результат хуже, чем метод, основанный на вычислении спектра сигнала. Анализируя матрицу ошибок, можно заметить, что классы «blink» и «double blink» плохо разделимы, поскольку как при моргании, так и при двойном моргании сигналы с электродов коррелируют одинаково, разница лишь в том, что второй сигнал содержит два пика. Информация об этом при подсчёте корреляции, как можно заметить, почти

не учитывается. При этом сигналы с морганием левым и правым глазом лучше различает method 1. Также данный метод лучше различает моргания одним и двумя глазами. Таким образом, данный метод можно применить для классификации на три класса: «left wink», «right wink» и «blink», и он должен показать высокое качество.

Перспективными направлениями дальнейших исследований являются:

- уменьшение времени между соседними классификациями сигнала в игре для повышения отзывчивости управления;
- добавление возможности управления скоростью объекта, анализируя различные ритмы;
- усложнение правил игры при помощи сигналов, базирующихся на представлении движений.

Заключение

В ходе данного исследования была реализована игра «Лабиринт», управление в которой основывается на морганиях. Была показана принципиальная возможность и эффективность корреляционного анализа для классификации различных типов морганий, а именно, были предложены методы формирования признакового пространства, основанные на корреляционной функции, для классификации морганий на три класса: моргание левым, правым глазом и отсутствие моргания. Достигнуто качество классификации порядка 97%.

Литература

1. *Rajesh Singla, Brijil Chambayil, Arun Khosla, Jayashree Santosh.* Comparison of SVM and ANN for classification of eye events in EEG — J. Biomedical Science and Engineering, 4 (2011), pp 62-69.
2. *S. Rihana, P. Damien, T. Moujaess.* EEG-Eye Blink Detection System for Brain Computer Interface — Converging Clinical and Engineering Research on Neurorehabilitation Biosystems & Biorobotics Volume 1 (2013), pp 603-608.
3. *Jorge Baztarrica Ochoa.* EEG Signal Classification for Brain Computer Interface Applications (2002).
4. *Воронцов К.В.* Математические методы обучения по прецедентам (теория обучения машин). [HTML] (<http://www.machinelearning.ru/wiki/images/6/6d/Voron-ML-1.pdf>)

5. *Finke A. Lenhardt A. Ritter H.* The MindGame: A P300-based brain-computer interface game. — *Neural Networks*, 22 (2009), pp 1329-1333.
6. *Кнут Д., Грэхем Ф., Поташник О.* Конкретная математика. Основание информатики. — М.:“Мир, 1998.
7. <http://www.csie.ntu.edu.tw/~cjlin/liblinear/>
8. <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>
9. *Breiman, Leo.* Random Forests. — *Machine Learning* (2001), 45(1), 5–32.
10. *Freund Y., Schapire R., Abe N.* A short introduction to boosting — *Journal-Japanese Society For Artificial Intelligence*. — 1999. — Т. 14. — №. 771-780. — С. 1612.