

Санкт-Петербургский государственный университет  
Математико-механический факультет

**Обучение классификаторов жестов мышью  
с помощью алгоритма AdaBoost в проекте QReal**

Курсовая работа студентки 245 группы  
Занько Софьи Владимировны

Научный руководитель ст. преп.  
Литвинов Юрий Викторович

Санкт-Петербург  
2012

## Оглавление

Введение .....	2
Алгоритм AdaBoost.....	2
Детали реализации .....	3
Тестирование.....	4
Заключение .....	5
Список литературы.....	6

## Введение

Средство QReal, разрабатываемое на кафедре системного программирования СПбГУ, представляет собой metaCASE-систему, или среду разработки, позволяющую создавать новые визуальные языки программирования и использовать их для написания пользовательских программ. Эти программы представляются в виде диаграмм, элементами которых являются объекты из определённого для данного языка списка. В ходе программирования пользователю необходимо постоянно обращаться к списку, находящемуся в специальном окне приложения, и при помощи операции drag-and-drop перетаскивать объекты на рабочее поле. Для упрощения и ускорения процесса в Qreal была встроена система распознавания жестов мышью, которая позволяет добавлять объекты при помощи движений мыши, образующих команды.

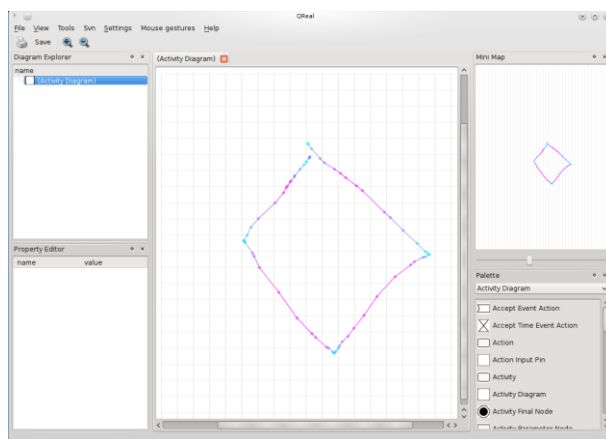


Рисунок 1. Рисование жеста Decision Node диаграммы активности UML

Алгоритмы, определяющие, какому классу объектов принадлежит изображённый пользователем жест, называются классификаторами. Ранее в проекте были испробованы различные их виды, но все они показывали неудовлетворительную точность распознавания, равную, в лучшем случае, 76%. Действенным способом улучшить точность являются алгоритмы обучения, корректирующие идеальные жесты для распознаваемых объектов на основе данных обучающей выборки. Проблема состоит в том, что для вновь добавленных жестов подобные алгоритмы будут абсолютно бесполезны. В связи с этим возникает задача поиска не изменяющего идеальные жесты алгоритма обучения, который, возможно, улучшит результаты распознавания вновь добавленных жестов.

Жесты, соответствующие объектам визуального языка, можно условно разделить на простые и сложные, при этом сложные жесты могут «содержать» в себе один или несколько простых. К примеру, в языке графического описания для объектного моделирования UML объект Actor представляет собой фигуру человека с головой в виде окружности, которая сама по себе является объектом Initial Node. В связи с этим, была выдвинута гипотеза о том, что обучение классификаторов на наборе простых жестов может улучшить качество распознавания, в том числе, и сложных жестов.

## Алгоритм AdaBoost

Алгоритмом, удовлетворяющим поставленным в начале требованиям, является AdaBoost (сокр. Adaptive Boosting), предложенный в первоначальном варианте в

1997 г. Йоавом Фройндом (Yoav Freund) и Робертом Шапире (Robert Shapire). Идея алгоритма состоит в том, чтобы объединить несколько слабых классификаторов в линейную комбинацию, представляющую собой более сильный классификатор.

На вход алгоритму подаётся список классификаторов и некоторое обучающее множество, состоящее, в нашем случае, из пар вида *название объекта – жест пользователя*. Далее запускается цикл, последовательно вызывающий все классификаторы. После попытки распознавания классификатором всех объектов обучающего множества, вычисляется ошибка классификации, на основе которой классификатор получает определённый вес, тем больший, чем меньше полученная ошибка. Затем происходит перераспределение весов объектов обучающего множества, отвечающих за важность данного объекта для классификации. Объекты, неверно классифицированные предыдущим классификатором, получают больший вес, что заставляет «сфокусироваться» на них следующие классификаторы.

Первоначальный вариант алгоритма AdaBoost рассчитан на бинарные классификаторы, то есть такие, которые классифицируют объект в один из двух имеющихся классов. В нашем же случае классов обычно оказывается гораздо больше. Существуют различные расширения алгоритма на случай нескольких классов, из которых нами был выбран AdaBoost.M1, представляющий собой прямое расширение AdaBoost, поскольку этого достаточно для проверки сформулированной гипотезы.

## Детали реализации

Внедрение AdaBoost в Qreal потребовало некоторых изменений в архитектуре проекта. На начальный момент реализованные в проекте алгоритмы распознавания являлись объектами класса *GesturesManager*, обладающего методом *getDistance()*, возвращающим расстояние от данного жеста до некоторого идеального. Также в проекте был класс *AbstractRecognizer*, заключающий в себе общую логику распознавания, имевший поле типа *GesturesManager* и метод *recognizeObject()*, возвращавший такой объект, что расстояние от его идеального жеста до жеста, изображённого пользователем, минимально. Формально результатом работы AdaBoost должен являться классификатор, то есть объект типа *GesturesManager*, но на практике класс *AdaBoostManager*, заключающий в себе алгоритм, работает как *AbstractRecognizer*, поскольку никакой формулы для вычисления расстояния между жестами он не предоставляет. В связи с этим, общая функциональность *AdaBoostManager* и *AbstractRecognizer* была вынесена в абстрактный класс *RecognizerInterface*. При этом была сохранена возможность при тестировании выбирать для распознавания по необходимости любой из ранее реализованных классификаторов.

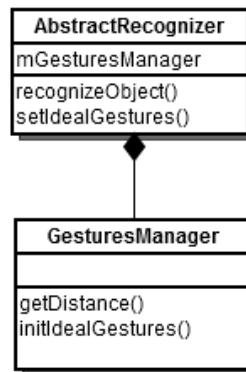


Рисунок 2. Исходная структура

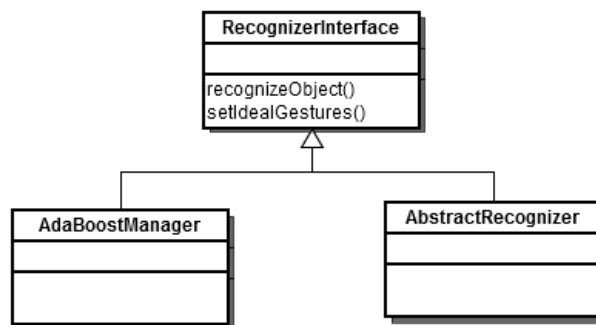


Рисунок 3. Структура после рефакторинга

В ходе встраивания алгоритма в проект был проведён также некоторый рефакторинг кода, заключающийся, в основном, в изменении типов данных и подгонке кода под стайлгайд проекта.

## Тестирование

Апробация алгоритма производилась на тестовой выборке, включавшей в себя элементы диаграммы активностей языка UML 2. Данная тестовая база была получена с помощью добровольцев, которым было предложено повторить жесты из списка, включавшего 9 элементов.

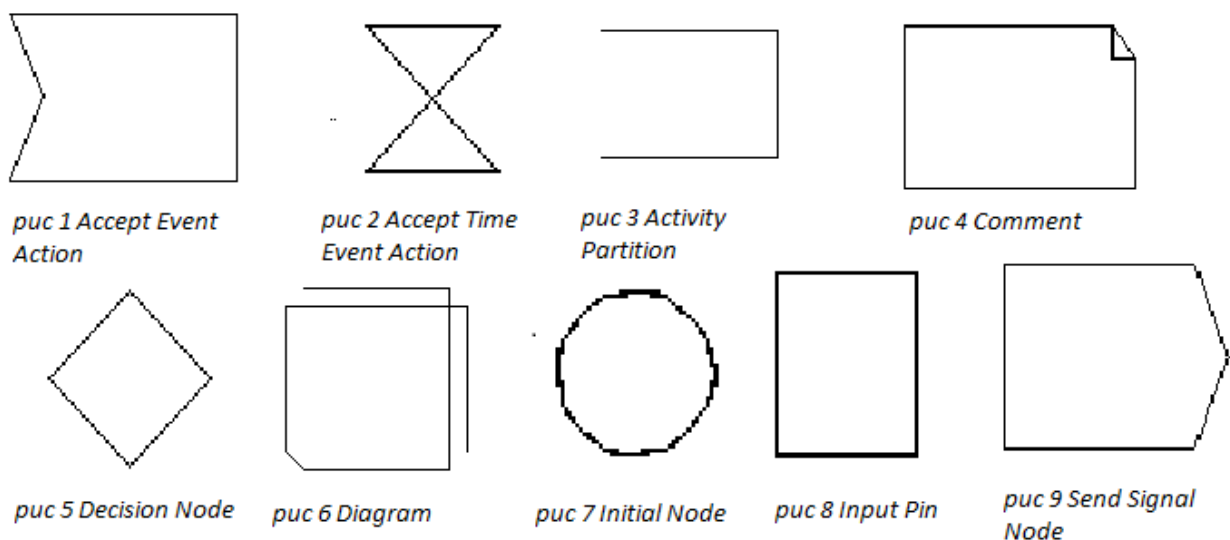


Рисунок 4. Объекты базы жестов пользователя

В качестве обучающих жестов были выбраны 4: Activity Partition, Decision Node, Initial Node, Input Pin. Всего в обучающей выборке было 558 жестов. В тестовую же выборку попало, в целом, 628 жестов, которые принадлежали к пяти классам: Accept Event Action, Accept Time Event Action, Comment, Diagram и Send Signal Action. Ниже представлены результаты распознавания алгоритмом AdaBoost и отдельным классификатором, показавшим наилучшую точность в предыдущих тестированиях.




	Объект					Всего расп.	Всего жестов	%
								
AdaBoost	114/115	137/142	9/128	122/122	108/121	490	628	78
Отдельный классификатор	112/115	137/142	7/128	122/122	107/121	485	628	77

Таблица 1. Результаты тестирования

Как видно из таблицы, полученное улучшение в распознавании составляет 1%. Этот результат мы вынуждены признать неудовлетворительным.

В данном тестировании результирующий классификатор AdaBoost был получен с помощью четырёх классификаторов, показавших лучшие результаты в предыдущих экспериментах. При незначительном улучшении в точности, это привело к заметному замедлению работы алгоритма распознавания, что также следует отнести к недостаткам использования AdaBoost.

Стоит отметить, что при тестировании алгоритма на тестовой выборке, включавшей в себя, в том числе, и жесты, на которых проводилось обучение алгоритма, результирующая точность составила 83% (обучающая выборка – 20 жестов, 4 объекта, тестовая выборка – 934 жеста, 7 объектов, использовано 3 классификатора), то есть было получено существенное улучшение точности. Отсюда можно сделать вывод о том, что AdaBoost является эффективным обучающим алгоритмом, но не применим при генеративном подходе к созданию жестов.

## Заключение

Итогом проделанной работы является алгоритм AdaBoost, встроенный в инструмент распознавания жестов системы QReal. Выдвинутая нами в начале гипотеза о том, что классификатор, полученный с помощью AdaBoost, будет более точно распознавать сложные жесты после обучения алгоритма на наборе простых жестов, не подтвердилась. Полученное улучшение в распознавании при учёте общего замедления работы признано неудовлетворительным.

В ходе работы были также проведены незначительные рефакторинг и оптимизация существующего кода проекта.

## Список литературы

- 1) Maria Osechkina, Yuri Litvinov, Timofey Bryksin. Multistroke Mouse Gestures Recognition in QReal metaCASE Technology // SYRCoSE 2012: Proceedings of the 6th Spring/Summer Young Researchers' Colloquium on Software Engineering. pp. 194-200
- 2) Ji Zhu, Saharon Rosset, Hui Zou, Trevor Hastie, Multi-class AdaBoost// URL <http://www.stanford.edu/~hastie/Papers/samme.pdf> (Дата обращения 13.05.2012)
- 3) Robert E. Shapire, Yoram Singer. Improved Boosting Algorithms Using Confidence-rated Prediction// Machine Learning, 37, 297–336 (1999)