

Приложение А. Описание работы программы

А.1. Основные положения

А.2. Рабочие области программы

А.3. Дальнейшие разработки

А.1. Основные положения

Прежде чем перейти к непосредственному описанию взаимодействия с программой немного рассмотрим основные моменты организации ее интерфейса, а также внутренней структуры программного модуля.

Для облегчения написания программы, с возможностью ее быстрого дополнения в последующем было принято решение разбить основной программный модуль на несколько вспомогательных, каждый из которых будет содержать в себе функции и методы для работы каждой отдельной ячейки программы. Задача же основного модуля, в таком случае, будет состоять в установке связей между модулями, а также организации удобного, с точки зрения пользователя, доступа.

Кратко перечислим данные модули:

- модуль хранения основных данных сети, в который занесена текущая структура сети и ее характеристики в виде описанных ранее в п.п. 4.3 экземпляров объектов;
- модуль организации двухмерной графики, содержащий в себе основные графические функции, требуемые для подготовки и получения входных изображений;
- модуль организации трехмерной графики, с помощью которого происходит отображение наглядной информации, несущей в себе данные, о текущих процессах в сети и результатах ее работы;
- основной модуль, выполняющий функции инициализации дополнительных модулей, организации интерфейса, осуществления инициализации нейронной сети, а также доступа к Базам Данных.

Подобный способ программирования не является какой-либо новой разработкой, такие методы были предложены давно и в настоящее время с успехом используются.

Так как сама программа является многофункциональной, то при ее разработке возникла проблема организации удобного интерфейса, вследствие некоторой перегруженности основного модуля стандартными компонентами, одновременное появление которых на экране монитора могло бы вызвать некоторое замешательство пользователя. Вторым фактором, на который был сделан акцент при подготовке интерфейса стало то, что между основными действиями программы, как то: подготовка входных изображений, обучение сети, установка задачи на распознавание, должны проходить не одновременно, а в некоторой, довольно жесткой последовательности, нарушение которой, может привести к негативным результатам. При этом стоит учесть, что пользователю совершенно не интересно, каким образом происходит распознавание, используется ли при этом нейронная сеть, или же специальные математические алгоритмы, важен результат, и, поэтому изображать на экране каким-либо образом сам процесс распознавания не имеет смысла. И, наконец, последним учтенным фактором стало то, что многооконный интерфейс также не способствует повышению наглядности, поэтому стоило бы рассмотреть возможность организации расположения внешней области программного модуля в одном окне, которое могло бы в некоторой степени видоизменяться в зависимости от происходящих в данный момент процессов.

На основании вышесказанного было принято решение об использовании единого окна программы, состоящего из четырех основных страниц, появление одной из которых скрывало бы информацию на остальных, в данный момент не используемых, страницах. При этом организация единого меню, соединяющего между собой информацию, принадлежащую каждой странице в отдельности, и всей программе

одновременно, будет происходить в специально выделенной панели, расположенной в левой области окна.

Рассмотрим каждую страницу более подробно.

А.2. Рабочие области программы

А.2.1. Страница подготовки входных изображений

Естественно было бы предложить, что первая рабочая страница, к которой должен был бы обратиться пользователь после запуска программы, должна содержать в себе компоненты подготовки входных изображений (рис. А.1).

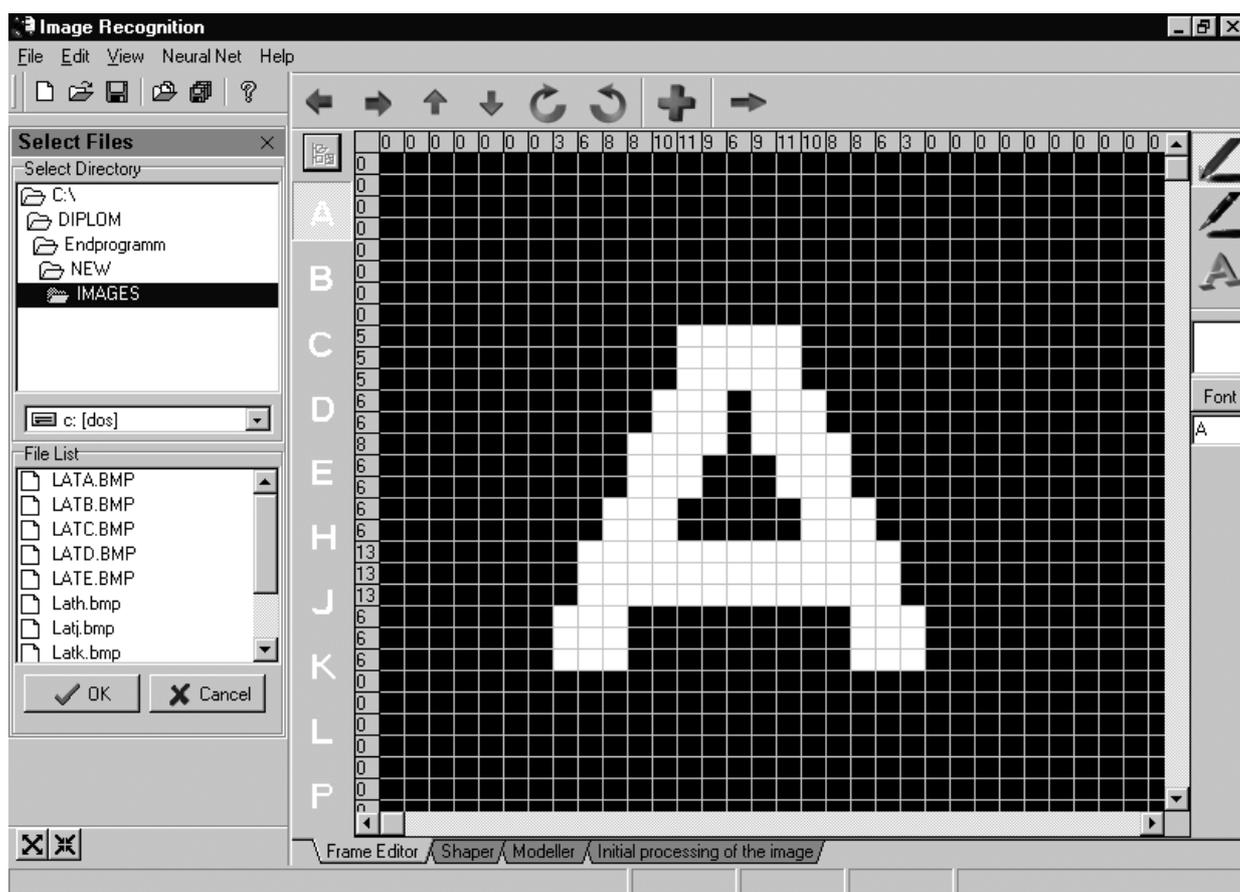


Рис. А.1

Как видно из рисунка в левой части текущего окна расположено меню выбора рабочего каталога, в котором находятся файлы изображений. Как уже отмечалось ранее, данные файлы должны иметь разрешение 32×32 пиксела, и иметь графический формат BMP. Выбрав текущий каталог в листе (File List), расположенном немного ниже диалога выбора каталога (Select Directory) можно увидеть список открытых файлов нужного формата. После нажатия клавиши ОК, данный список будет открыт для выбора в левой части открытой страницы (Frame Editor). На рисунке А.1 изображен момент выбора изображения буквы “А”. После загрузки изображения, при помощи панелей, изображенных на рис. А.2 можно произвести его редактирование. Стоит заметить, что на вертикальной и горизонтальной панелях, окружающих увеличенную сетку с текущим

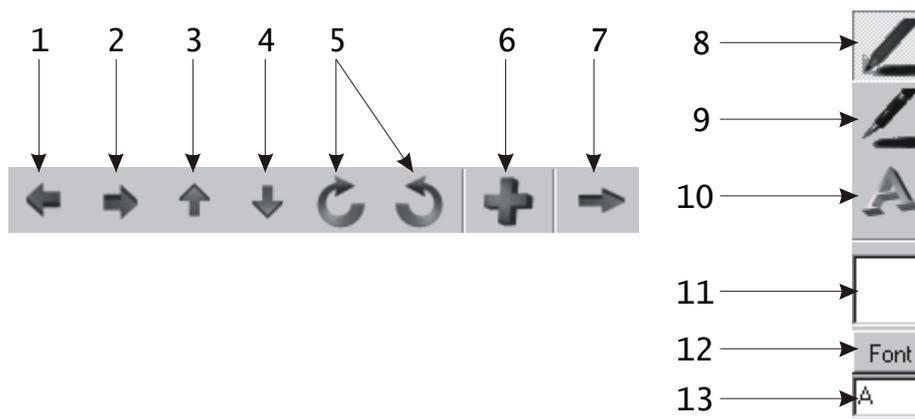


Рис. А.2

изображением, отмечены суммы пикселей белого цвета по столбцам и строкам соответственно.

Рассмотрим назначение кнопок на панелях редактирования изображений:

1. сдвиг изображения на один пиксел влево;
2. сдвиг изображения на один пиксел вправо;
3. сдвиг изображения на один пиксел вверх;
4. сдвиг изображения на один пиксел вниз;
5. вращения изображения по- и против часовой стрелки, в данной версии программы не имеют актуального значения;
6. активизация функции избавления от единичных помех в изображении;
7. экспорт обработанного изображения во входной вектор, для одачи на вход нейронной сети;
8. метод рисования ручкой текущим цветом;
9. метод рисования кисточкой с заполнением закрашиваемой области текущим цветом;
10. рисование текста текущим цветом и установленным шрифтом;
11. панель смены текущего цвета, смена производится двойным нажатием клавиши мыши на саомй панели;
12. установка текущего шрифта;
13. редактирование текущего текста.

После окончания редактирования, изображение с помощью кнопки 7 может быть перекодированно в формат, нужный для подачи данного изображения на вход нейронной сети. При перекодировке также производится первичная обработка изображения, целью которой является выделение из изображения самого объекта инвариантно к его местоположению в изображении. Изображение может быть записано на диск, также в формате BMP. Стоит заметить, что непосредственно само редактирование можно производить в любом графическом редакторе, поддерживающем стандартные форматы, наиболее наглядным примером которого послужил бы редактор, входящий в поставку вместе с операционной системой Windows 95 и выше, называемый Paint. Но, так как при создании программы преследовалась идея поместить все требуемые для нормальной работы компоненты, то было решено поместить в сам модуль несложный редактор, который смог бы удовлетворить все требования, возникающие при обработке изображения.

После перекодировки изображения в вектор, следующим шагом, при работе с программой, будет подача вектора на вход сети и проведение операции обучения, либо распознавания. Что и будет происходить на следующей странице.

А.2.2. Страница активизации алгоритмов работы НС

На данной странице размещены компоненты, используемые для активизации основных функций, производимых над нейронной сетью. Перед рассмотрением интерфейса данной страницы стоит отметить, какие именно функции должны быть на ней размещены:

- *инициализация*, восстановление архитектуры сети после первой загрузки программы, посредством чтения информации из Базы Данных, и на основе этой информации восстановления количества нейронов во входном слое сети, а также значений весов для каждого нейрона;
- *обучение*, проведение операции чтения объекта из изображения, полученного после редактирования на предыдущей странице, увеличения количества нейронов во входном слое на единицу, и обучения данного нейрона;
- *распознавание*, чтение объекта из изображения, и сопоставление ему одного из объектов, которым сеть была ранее обучена (объект также может быть не распознан сетью, если значения выходов всех нейронов, после подачи на вход сети изображения, не превышает 0.7).

Вид данной страницы показан на рис. А.3.

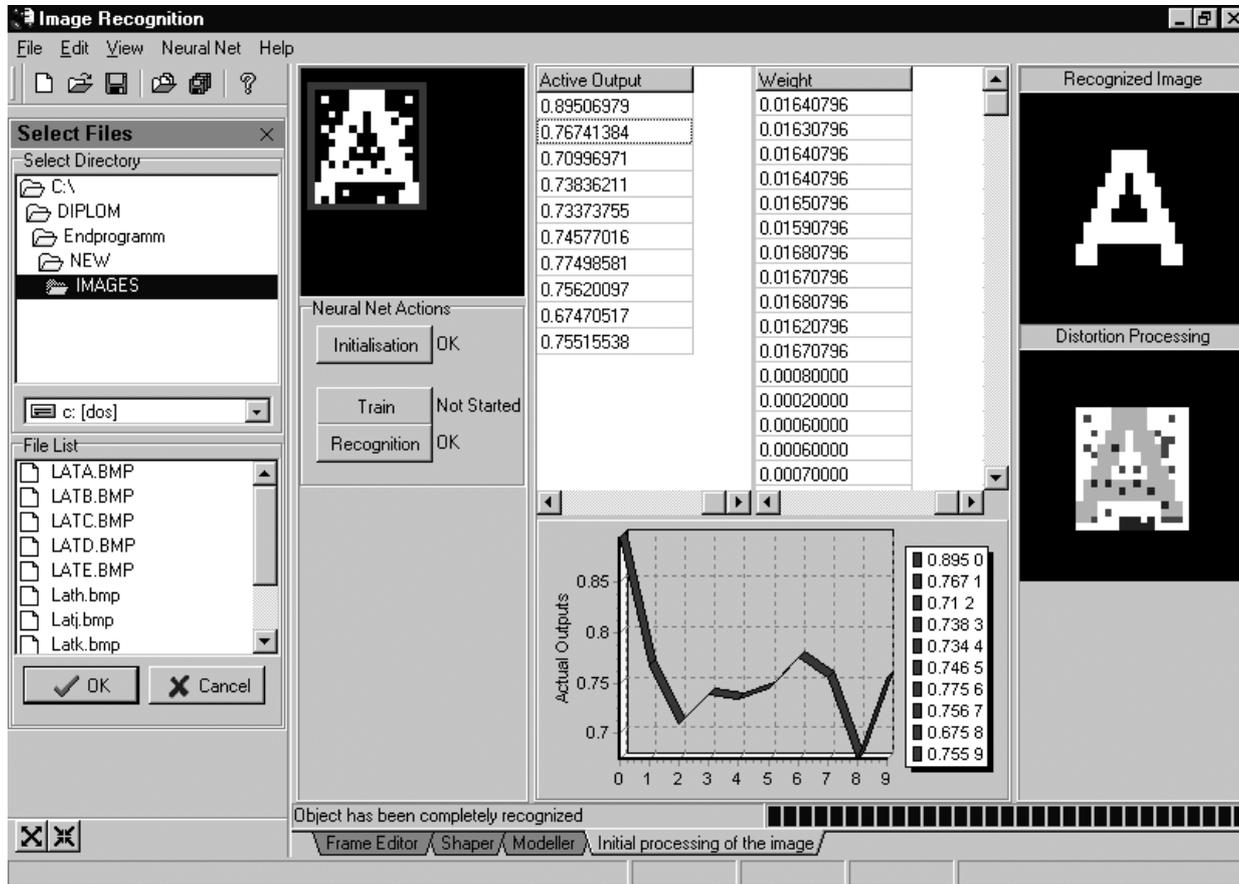


Рис. А.3

Перекодированное изображение, полученное из редактора, изображено в левом верхнем углу активной страницы. Как видно из рисунка изображение имеет помехи, однако после перекодировки сам объект уже выделен из изображения и обведен рамкой. Ниже изображения расположены три кнопки, через которых активизируются функции сети, описанные ранее. Данные о текущем состоянии сети расположены в середине страницы в виде двух таблиц и графика. В первой таблице изображены текущие состояния выходов нейронов сети, та же информация, но в графическом виде, для повышения наглядности, изображена на графике, расположенном внизу страницы. Во второй таблице, для более подробного ознакомления с результатами обучения сети можно ознакомиться с значениями весов для каждого нейрона сети.

В правой части страницы изображены результаты распознавания, верхнее изображение показывает объект, сопоставленный в результате распознавания входному изображению (Recognized Image), а на нижнем демонстрируется анализ помех (Distortion Processing), который производится после распознавания на основе данных об объекте, запомненном сетью. В результате анализа из изображения выделяется сам объект и сравнивается с сопоставленным, те пиксели, значения которых считаются ошибочными в результате сравнения, показываются более темным цветом. В качестве применения данной операции анализирования помех в изображении можно, например, отметить проблему восстановления испорченных изображений, что является в настоящее время одним из актуальных вопросов.

В заключение можно было бы отметить, что на рис. А.3 изображен момент распознавания буквы «А» и последующего анализа помех во ее изображении.

А.2.3. Страницы подготовки изображений высокого порядка сложности

Основным предназначением разрабатываемой сети не является само распознавание букв латинского алфавита или же подобных простых объектов. Реализованный и описанный алгоритм распознавания является всего лишь примером работы сети и в настоящем виде не может быть применен в заводских условиях. Однако стоит заметить, что принцип распознавания, реализованный в настоящее время может с успехом применяться и для гораздо более сложных изображений. Остается только разработать принцип их получения. Если применить стандартный метод съемки детали на камеру, а затем перекодировки полученной фотографии во входной вектор, то в таком случае очень трудно будет избавиться от помех, которые обязательно будут сопутствовать процессу такого рода. То есть идеальное изображение объекта, по которому сеть должна обучаться, уже имеет помехи, распознать которые сеть не сможет, так как эталонного изображения в ее память еще не занесено. Таким образом, казалось бы простой вопрос, несет за собой целый ряд проблем, решение которых в настоящее время до сих пор еще не найдено.

В настоящей дипломной работе предлагается метод получения эталонных изображений деталей непосредственно во время их проектирования. Современные методы проектирования подразумевают под собой обязательное участие компьютера в самом процессе разработки. Например, наиболее мощной в настоящий момент системой автоматического проектирования на базе персональных ЭВМ является AutoCAD, в возможности которого входит как разработка конструкторских чертежей деталей, так и получение их трехмерного изображения. Следовательно, если каким-либо образом получить изображение детали непосредственно из самой системы проектирования, то изображение будет идеальным, то есть не иметь помех, что убережет нас от излишних ошибок.

Таким образом, разработав требуемый алгоритм, можно будет с уверенностью заявить, что нейронные сети могут с успехом применяться в СТЗ для распознавания изображений, и превзойдут в данном вопросе современные математические алгоритмы распознавания изображений. На последующих двух страницах программы предлагаются два метода получения такого рода изображений.

На первой странице предлагается метод получения изображения детали из конструкторских чертежей, которые программа может получить из наиболее распространенного в настоящее время формата файлов DXF, наиболее часто используемом в системе автоматического проектирования AutoCAD (рис. А.4).

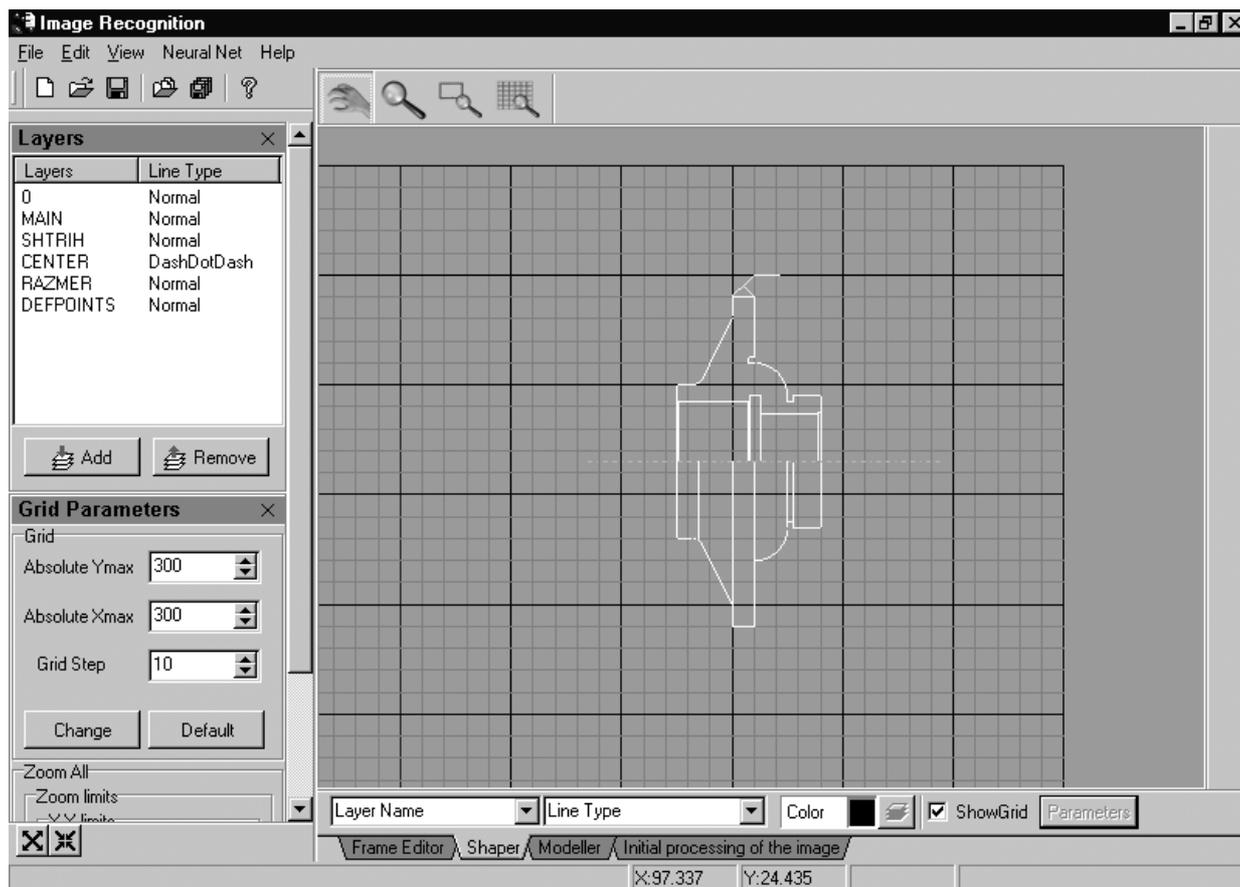


Рис. А.4

Принцип подготовки изображений на базе конструкторских чертежей основан на получении контура детали по полилинии, огибающей внешнюю поверхность ее корпуса. Внутренняя часть (принадлежащая детали) закрашивается белым цветом, внешняя – черным. В представленном на рис. А.4 примере был получен контур детали вращения, спроектированной в системе AutoCAD, на основании которого можно провести подобную операцию. Формат DXF поддерживает многослойность изображения, что также можно использовать в работе по выделению контура детали (список текущих слоев для данной детали можно видеть в меню, в правой части рисунка). Непосредственно в самом редакторе можно настроить положение детали, при котором пользователь хочет получить требуемое положение контура в изображении, для этого служит панель, расположенная сверху текущей страницы. Сетка служит для получения точных значений координат при проведении операции пересчета систем координат детали и камеры. Сетку

можно настроить с помощью диалога, вызываемого нажатием кнопки Parameters, в нижней рабочей панели страницы.

Однако такой способ получения сложных изображений может быть в недостаточной степени точным для более сложных деталей. Даже в таком простом случае, когда в наличии имеется деталь вращения, нам может потребоваться несколько ее видов. Например деталь вращения в данном случае сверху выглядит как окружность, а сбоку она больше похожа на прямоугольник, что сеть может посчитать как два разных объекта. Причем не стоит забывать о возможных отверстиях в детали, которые могут быть видны не во всех видах, но иметь решающее значение при распознавании. Таким образом можно прийти к выводу о том, что двумерное изображение детали не всегда может представлять полную информацию для нейронной сети. То есть следует также разработать редактор, способный получать двухмерные изображения на основе трехмерного описания объекта. Универсальность формата DXF в вопросе хранения как двухмерной так и трехмерной информации оказала некоторую помощь в создании такого редактора. В системе проектирования AutoCAD можно получить трехмерное изображение детали, сохранить данное изображение в DXF формате, а затем прочитать его в специализированном редакторе, в котором можно произвести операцию тонирования, и получить несколько сложных изображений любого разрешения, содержащих в себе основные виды детали.

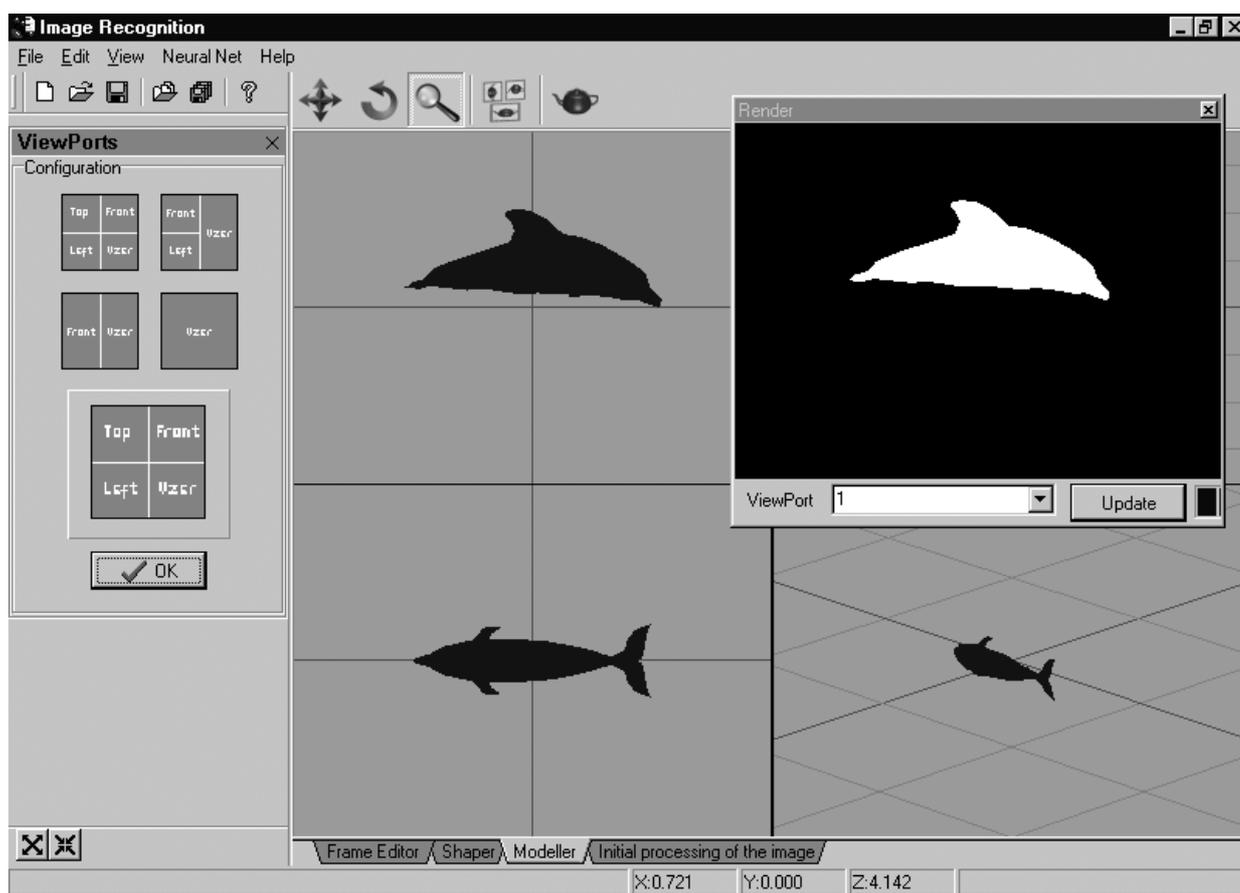


Рис. А.5

В разрабатываемую программу был включен модуль, реализующий в себе основные возможности такого редактора (рис. А.5). На рисунке изображена трехмерная модель дельфина, полученная в AutoCAD и экспортированная в трехмерный редактор. В котором с помощью управляющих клавиш, расположенных в верхней панели страницы

можно настроить требуемый вид объекта в текущем порту экрана. Которые также могут настраиваться при помощи диалога, расположенного в левом меню. После проведения операций редактирования и тонирования, полученные виды объекта могут быть сохранены в монохромном BMP формате, в требуемом разрешении. Сложность объекта в данном редакторе регламентируется только возможностями персонального ЭВМ, на котором происходит обработка изображений, и возможностями того редактора, в котором создается объект. При создании данного модуля не преследовалось цели создать функции для редактирования непосредственно самого объекта, так как работы в данном направлении ведутся уже более 20 лет, и современные редакторы обладают исчерпывающими возможностями в данном вопросе.

Таким образом в конечную программу были включены, помимо непосредственно модуля распознавания, основанного на алгоритмах работы нейронной сети, но также и два модуля получения эталонных изображений, облегчающих и ускоряющих процесс обучения НС.

А.3. Дальнейшие разработки

Стоит заметить, что в настоящее время данная программа не имеет коммерческой ценности, так как реализованные в ней алгоритмы распознавания являются только примером работы сети. Конечно в дальнейшем, на основе разработанных алгоритмов, при условии продолжения работы в данном направлении, в короткие сроки возможно создание коммерческого пакета, который можно было бы применять в заводских условиях.

Работы над модулями двух- и трехмерной графики в настоящее время еще не окончены, так как объем работы в данном направлении огромен, и эти разработки не являлись конечной целью дипломной работы. Однако, при соответствующей их доработке на основе выполненной работы, эти модули также могут быть включены в основной пакет.

Таким образом можно выделить три основных направления дальнейших разработок в данной области:

1. изменение структуры сети, с целью достижения возможности подачи входных изображений большего разрешения, чем в настоящий момент;
2. доработка модулей получения более сложных эталонных изображений;
3. увеличение количества цветов во входных изображениях, для получения более реалистичной картины об окружающей обстановке.

Целью создания программы было доказательство преимущества алгоритмов распознавания, основанных на методах нейронных сетей, перед современными алгоритмами, используемыми в настоящее время. По результатам работы программы можно сделать вывод, что никакая современная система распознавания не выдержит 40-процентного содержания шумов в исходном изображении. А следовательно, применение приведенных выше алгоритмов работы не только возможно, но и способно решить многие современные проблемы, которые стоят перед учеными в вопросах разработки искусственного интеллекта.