

## **ZASTOSOWANIE SZTUCZNYCH SIECI NEURONOWYCH DO ROZPOZNAWANIA METALI NA PODSTAWIE WYKRESU STATYCZNEJ PRÓBY ROZCIĄGANIA**

**Dawid Ewald**

*Uniwersytet Kazimierza Wielkiego  
Instytut Techniki  
doktorant  
ul. Chodkiewicza 30/ p.215, 85-064 Bydgoszcz  
e-mail: dawidewald@gmail.com*

**Streszczenie:** *W artykule przedstawia zagadnienie sztucznych sieci neuronowych oraz ich wykorzystania w klasyfikacji metali na podstawie wykresu statycznej próby rozciągania. W pracy opisano działanie sieci neuronowych oraz sposób ich wykorzystania..*

**Słowa kluczowe:** *Sztuczne sieci neuronowe, FANN, Delphi*

### **Application of artificial neural networks for recognition of metals on the basis of static tensile test chart**

**Abstarct:** *In this article presents the issue of artificial neural networks and their use in the classification of metals on the basis of the static tensile test chart. This paper describes the operation of neural networks and how to use them.*

**Keywords:** *Artificial neural networks, FANN, Delphi*

#### **1. WSTĘP**

Współczesny naukowiec ma do dyspozycji wiele narzędzi ułatwiających pracę naukową. Wiele z tych narzędzi posiada ogromną bazę danych pozwalającą użytkownikowi skrócić procesy badawcze oraz wygodniej i szybciej zarządzać zgromadzonymi materiałami. Narzędzia takie często opierają się na algorytmach działania co ogranicza ich skuteczność. Techniki sieci neuronowych pozwalają nam tworzyć narzędzia pozbawione tego ograniczenia a zarazem szybko rozwijać nowe sposoby ich zastosowania w tych sytuacjach, w których mamy do czynienia z nieliniowymi problemami.

#### **2. BIOLOGICZNE PODŁOŻE BUDOWY SZTUCZNEGO NEURONU**

Komórka nerwowa zwana potocznie neuronem, stanowi podstawowy element systemu nerwowego organizmu. Neuron składa się z ciała zawierającego jądro oraz odchodzących od niego licznych wypustek, te o strukturze drzewiastej zwane dendrytami (greckiego słowa "déndron", czyli drzewo) oraz grubsze rozwidlający się na końcu aksony.

W neuronie biologicznym sygnał wejściowy doprowadzany jest za pośrednictwem dendrytów, zaś sygnał wyjściowy odprowadzany jest za pomocą aksonu. Upraszczając system działania tak by zilustrować przepływ sygnałów można przyjąć, że przekazywanie impulsu od jednego neuronu do następnego odbywa się za pośrednictwem neuroprzekaźnika

wydzielanego pod wpływem bodźców pochodzących od synaps.

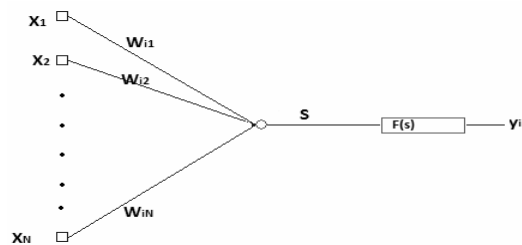
Analogicznie sztuczny neuron zbudowany jest z wejść ( $x_1, x_2 \dots x_n$ ) (biologiczne dendryty), odpowiednim im współczynnikiem wagowym ( $w_{i1}, w_{i2} \dots w_{in}$ ) oraz elementu sumującego co przedstawiono na rysunku 2.1.

Sposób przetwarzania informacji wejściowych  $x$  na informację wyjściową  $y$  uzależniony jest od mogących się zmieniać podczas uczenia współczynników wagowych. Sztuczny neuron mnoży poszczególne sygnały wejściowe  $x$  przez odpowiadające im współczynniki wag  $w$  a następnie je sumuje. Sygnał wyjściowy może przyjmować 0 lub 1 zgodnie z zależnością:

$$y_i(s_i) = \begin{cases} 1, & s_i > 0 \\ 0, & s_i \leq 0 \end{cases} \quad (1)$$

$$s_i = \sum_{i=1}^m w_i X_i \quad (2)$$

Gdzie:  $S_i$  to sygnał sumatora opisany wzorem 2.



Rysunek. 2.1 Schemat budowy sztucznego neuronu.

### 3. JAK DZIAŁA SIĘĆ NEURONOWA

Jak wynika z budowy pojedynczego neuronu posiada on swoją pewną wewnętrzną pamięć reprezentowaną poprzez wartość wag i progę. Jednak jego możliwości przetwarzania sygnału są bardzo ograniczone co powoduje małe zapotrzebowanie na moc obliczeniową. Tworząc z pojedynczych neuronów sieć można zbudować system realizujący bardzo złożone zadania. Dlatego sieć neuronowa musi działać jako całość. Działanie sieci polega na przesyłaniu między neuronami kolejnych warstw sygnałów od wejścia pierwszej warstwy do wyjścia ostatniej. Sygnał w poszczególnych neuronach jest przetwarzany i przekazywany do kolejnego.

Najważniejsza cecha neuronowych jest możliwość uczenia. Istnieje wiele sposobów uczenia zależnych od

modelu sieci oraz problemu, który przednią stawiamy. Najczęściej stosowany sposób to uczenie z nauczycielem.

Polega ono na podaniu konkretnego sygnału wejściowego i wyjściowego, pokazując w ten sposób jaka jest wymagana odpowiedź. Dane te zwane są danymi uczącymi i muszą zostać odpowiedni dobrane aby skrócić proces uczenia sieci.

### 4. CHARAKTERYSTYKA STATYCZNEJ PRÓBY ROZCIĄGANIA

Próba statycznego rozciągania jest podstawową i najczęściej stosowaną próbą wytrzymałościową, jednak trzeba mieć na uwadze fakt, że wielkości charakterystyczne uzyskane podczas przeprowadzania tej próby nie powinny być uznawane jako odzwierciedlenie ogólnego zachowania się konstrukcji pod obciążeniem. Dlatego niektóre elementy tj. liny, których obciążenie robocze to rozciąganie, poddaje się próbie rozciągania w całości .

Statyczna próba rozciągania metali ujęta jest normą polega na poddaniu działaniu siły rozciągającej w kierunku osiowym na odpowiednio przygotowaną próbkę aż do jej zerwania. Mimo iż próbę rozciągania nazywa się statyczną, to jednak obciążenie wolno narasta z określoną prędkością. Przy czym zakłada się, że odpowiadające w stanie spoczynku badanym naprężeniem odkształcenia, pojawiają się natychmiast po przyłożeniu obciążenia, tzn., że istnieje w każdej chwili przeprowadzania badania równowaga naprężenia i odkształcenia. Takie założenie jest słuszne dla odkształceń sprężystych, natomiast w zakresie odkształceń plastycznych dla wielu materiałów przyjęcie takiego założenia jest niezgodne z rzeczywistością. Maksymalny przyrost naprężeń jest określony normą i nie powinien przekraczać 30 MPa/s oraz narastanie obciążeń powinno być ciągłe do swojej maksymalnej wartości.

Maszyna służąca do przeprowadzania próby rozciągania zwana jest zrywarką. Próbki do przeprowadzania badania posiadają część pomiarową o stałym przekroju i są zakończone główkami o zwiększonych wymiarach. Długość próbki musi być dobrana zgodnie z normą tak by przejściu do główek było łagodne co pozwala przyjąć, że stan odkształcenia i naprężenia w każdym punkcie części pomiarowej próbki jest jednorodny. W tak przyjętych warunkach można z pomiarów odkształceń na powierzchni ciała wyciągać wnioski o odkształceniach wewnątrz ciała, a z pomiarów siły można obliczyć naprężenia wewnątrz próbki .

W normie zostały zawarte następujące wielkości określające własności plastyczne i mechaniczne materiału:

- Średnica początkowa próbki (do [mm]) - średnica próbki na jej długości roboczej mierzona przed rozerwaniem.

- Średnica końcowa próbki ( $d_u$  [mm]) - średnica najmniejszego przekroju próbki w miejscu rozerwania.

- Średnica próbki do wyznaczania wydłużenia równomiernego

( $d_r$  [mm]) - średnica próbki po rozerwaniu mierzona na dłuższej części próbki w połowie odległości od miejsca jej rozerwania do końca długości pomiarowej.

- Długość pomiarowa początkowa ( $L_0$  [mm]) - długość odcinka badanej próbki, na której określa się wydłużenie.

- Długość próbki ( $L_t$  [mm]) - całkowita długość próbki.

- Długość pomiarowa końcowa ( $L_u$  [mm]) - długość pomiarowa próbki po rozerwaniu.

- Powierzchnia przekroju początkowego próbki ( $S_0$  [mm<sup>2</sup>]) - powierzchnia przekroju poprzecznego próbki na długości pomiarowej mierzona przed rozerwaniem.

- Powierzchnia przekroju końcowego ( $S_u$  [mm<sup>2</sup>]) - powierzchnia przekroju poprzecznego próbki w miejscu rozerwania.

- Bezwzględne wydłużenie próbki po rozerwaniu ( $\Delta L$  [mm]) obliczane na podstawie wzoru 4.1.

$$\Delta L = L_u - L_0 \text{ [mm]} \quad (4.1)$$

- Względne wydłużenie próbki proporcjonalnej po rozerwaniu ( $A_p$  [%]) obliczana ze wzoru 4.2:

$$A_p = \frac{\Delta L}{L_0} 100 \text{ [%]} \quad (4.2)$$

gdzie:

p - wskaźnik wielokrotności średnicy do lub wielokrotności  $\sqrt{S_0}$ .

- Względne wydłużenie równomierne próbki okrągłej ( $A_r$  [%]).

$$A_r = \frac{d_0^2 - d_r^2}{d_r^2} 100 \text{ [%]} \quad (4.3)$$

- Względne przewężenie próbki (Z [%]).

$$Z = \frac{d_0^2 - d_u^2}{d_r^2} 100 \text{ [%]} \quad (4.4)$$

- Względne przewężenie próbki okrągłej

$$Z = \frac{S_0 - S_u}{S_0} 100 \text{ [%]} \quad (4.5)$$

- Siła rozciągająca (F [N]) - siła działająca na próbkę w określonej chwili badania.

- Naprężenie rozciągające (R [MPa]) - naprężenie wyrażone stosunkiem siły F, do przekroju początkowego próbki  $S_0$ .

- Umowna granica sprężystości ( $R_{0,05}$  [MPa]) - naprężenie odpowiadające działaniu siły rozciągającej, wywołującej w próbce umowne wydłużenie trwałe x wynoszące 0.05% długości pomiarowej  $L_e$  wyrażane wzorem 4.6,

w technicznie uzasadnionych przypadkach dopuszcza się określenie granicy sprężystości przy wydłużeniach trwałych mniejszych niż 0.05%.

$$R_{0,05} = \frac{F_{0,05}}{S_0} \text{ [MPa]} \quad (4.6)$$

- Umowna granica plastyczności ( $R_{0,2}$  [MPa]) - naprężenie odpowiadające działaniu siły rozciągającej, wywołującej w próbce umowne wydłużenie trwałe x wynoszące 0.2% długości pomiarowej  $L_e$  wyrażane wzorem 4.7, w technicznie uzasadnionych przypadkach dopuszcza się określenie umownej granicy plastyczności przy innych wydłużeniach trwałych w granicach 0.05 - 0.5%.

$$R_{0,2} = \frac{F_{0,2}}{S_0} \text{ [MPa]} \quad (4.7)$$

- Siła odpowiadająca wyraźnej granicy plastyczności ( $F_e$  [N]) - siła przy której występuje wyraźny wzrost wydłużenia rozciąganej próbki; dla określonych materiałów rozróżnia się siłę  $F_{eH}$  odpowiadającą górnej granicy plastyczności oraz siłę  $F_{eL}$  odpowiadającą dolnej granicy plastyczności.

- Wyrażna granica plastyczności ( $R_e$  [MPa]) - naprężenia odpowiadające działaniu siły  $F_e$  obliczane ze wzoru 4.8:

$$R_e = \frac{F_e}{S_0} \text{ [MPa]} \quad (4.8)$$

Rozróżnia się górną granicę plastyczności  $R_{eH}$ , w której naprężenie odpowiada pierwszemu szczytowi obciążenia, zarejestrowanemu przy badaniu materiału oraz dolną granicę plastyczności  $R_{eL}$  odpowiadającą najmniejszej wielkości naprężenia przy wyraźnym wzroście wydłużenia; w przypadku, gdy występuje więcej niż jedno minimum pierwszego z nich nie bierze się pod uwagę.

- Największa siła ( $F_m$  [N]) - największa siła rozciągająca działająca na próbkę.

- Wytrzymałość na rozciąganie ( $R_m$  [MPa]) - naprężenie odpowiadające działaniu siły  $F_m$ .

$$R_m = \frac{F_m}{S_0} \text{ [MPa]} \quad (4.9)$$

- Siła rozerwania ( $F_u$  [N]) - siła rozciągająca w chwili rozerwania próbki.

- Naprężenie rozrywające ( $R_u$  [MPa]) - naprężenie odpowiadające działaniu siły  $F_m$ .

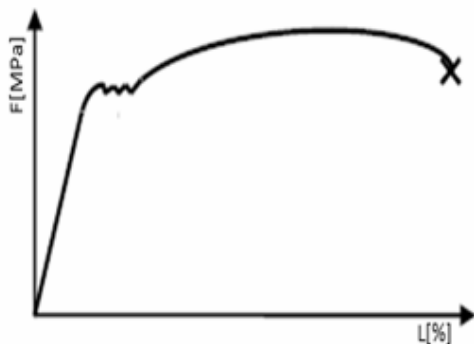
$$R_u = \frac{F_u}{S_0} \text{ [MPa]} \quad (4.10)$$

- Współczynnik sprężystości wzdłużnej ( $E$  [MPa]) - stosunek naprężenia  $R$  do odpowiadającego mu wydłużenia względnego  $A_p$  w zakresie, w którym krzywa rozciągania jest linią prostą.

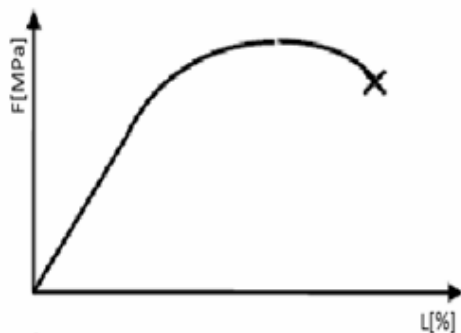
- Podatność maszyny ( $K$  [mm/N]) - stosunek zmiany odległości między uchwytami maszyny wytrzymałościowej do zmiany siły obciążającej.

- Powiększenie skali wydłużeń ( $\alpha$ ) - stosunek  $\Delta l$  odczytanego na wykresie do rzeczywistego  $\Delta l$  próbki.

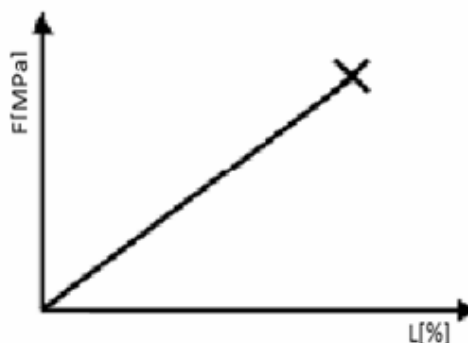
W wyniku przeprowadzania próby uzyskuje się wykres którego kształt zależy od cech plastycznych materiału. Można więc wyróżnić trzy podstawowe kształty wykresy:



Rysunek. 4.1 Materiał z wyraźną granicą plastyczności.



Rysunek. 4.2 Materiały bez wyraźnej granicy plastyczności.



Rysunek. 4.2 Materiał kruchy.

## 5. OPIS PROGRAMU ROZPOZNAJĄCEGO METAL NA PODSTAWIE WYKRESU STATYCZNEJ PRÓBY ROZCIĄGANIA

Sztuczne sieci neuronowe będą wykorzystywały darmową bibliotekę FANN w wersji 2.0 tworzoną przez Steffen Nissen. Biblioteka ta pozwala w prosty sposób implementować wielowarstwowe sieci neuronowe. Dostarcza również kilka metod uczenia oraz gotowy komponent możliwy do wykorzystania w kompilatorze Borland Delphi 7. Program będzie wykorzystywał cztery nauczone sztuczne sieci neuronowe. Pierwsza z nich zawierać będzie 122500 wejść oraz 3 wyjścia. Zadaniem tej sieci będzie rozpoznanie jednego z trzech kształtów wykresu. Danymi uczącymi były obrazy w formacie BMP o rozdzielczości 350 x 350 pikseli w tonacji monochromatycznej. Obrazy przedstawiają kształty wykresów statycznej próby rozciągania pozbawione osi. Podczas działania program zmienia obraz na postać 0 dla pikseli białych i 1 dla czarnych co znacznie ułatwia przetwarzanie go przez sieć. Tak przygotowane dane interpretowane są przez pierwszą sieć. Trzy kolejne sieci będą rozpoznawać metale podzielane na trzy grupy tj. pierwsza z sieci - metale z wyraźną granicą plastyczności, druga - metale bez wyraźnej granicy plastyczności oraz trzecia - metale kruche. Danymi uczącymi będą liczby odczytane z wykresów statycznej próby rozciągania. Uruchomienie odpowiedniej sieci będzie zależało od wartości na wyjściu sieci pierwszej rozpoznającej kształt wykresu. Obrazy zawierające kształty wykresów jak i same liczby opisujące te wykresy zostały stworzone do potrzeb eksperymentu.

**Czas na kod programu!**

Cały projekt składa się z trzech programów. Dlaczego aż trzech? Odpowiedź jest prosta. Ponieważ dwa pierwsze programy tworzą i trenują sieci neuronowe, które następnie wykorzystane są w trzeciej aplikacji.

#### Aplikacja „FANN - interpretacja wykresów”.

Zadaniem tej aplikacji jest stworzenie i wytrenowanie sztucznej sieci neuronowej rozpoznającej kształt wykresu statycznej próby rozciągania. Danymi uczącymi są obrazy w formacie 350 x 350. Aby możliwa była interpretacja przez sieć takich obrazów trzeba te obrazy zamienić na dane „rozumiane” przez sieć. W tym celu musimy obraz zamienić na tablicę zawierającą 0 i 1, gdzie 0 to kolor biały, a 1 to kolor czarny piksela.

Deklarujemy tablicę dwuwymiarową oraz jednowymiarową :

```
sgA: array [0..349, 0..349] of single;  
sgB: array [0..122499] of single;
```

następnie funkcję zamienObrazNaTablice realizującą „dekodowanie wykresu” oraz zapisanie wyniku do tablicy:

```
var  
i,j:integer;  
begin  
for i:=0 to 349 do  
begin  
for j:=0 to 349 do  
begin  
if  
GetRValue (image1.Canvas.Pixels [j,i])=0  
then  
begin  
sgA[i,j]:=1;  
end else  
sgA[i,j]:=0;  
end;  
end;  
end;  
end;
```

Kiedy mamy już wypełnioną tablicę dwuwymiarową danymi musimy ją przepisać do tablicy jedno wymiarowej ponieważ biblioteka FANN tego od nas wymaga zadanie to realizuje następująca funkcja przepiszTablice :

```
var  
i,j,m:integer;  
begin
```

```
m:=-1;  
for i:=0 to 349 do  
begin  
for j:=0 to 349 do  
begin  
m:=m+1;  
sgb[m]:=sga[j,i];  
end;  
end;  
end;
```

Kolejnym korkiem jest wygenerowanie pliku uczącego zawierające wszystkie dane uczące. Pierwszym krokiem jest stworzenie pliku na dysku z informacjami o ilości danych uczących oraz ilością wejść i wyjść sieci:

```
var  
TF : TextFile;  
begin  
AssignFile (TF,  
nazwapliku+'grupaMaterialow.data');  
If  
FileExists (nazwapliku+'grupaMaterialow.d  
ata') then  
// sprawdzenie, czy plik istnieje  
begin  
ReWrite (TF);  
// jeżeli nie - stwórz  
Writeln (TF, edit1.text);  
//dane do zapisu  
end else  
begin  
ReWrite (TF); // jeżeli nie - stwórz  
Writeln (TF, edit1.text);  
//dane do zapisu  
end;  
CloseFile (TF)  
end;
```

Po utworzeniu pliku czas na zapisanie wszystkich danych uczących:

```
var  
i, j, k: Integer;  
Plik: TextFile;  
begin  
AssignFile (Plik,  
nazwapliku+'grupaMaterialow.data');  
Append (Plik);
```

```

for k:=0 to listBox1.Count-1 do // w
listbox'i zapisane są ścieżki do obrazów
wykresów
begin
image1.Picture.LoadFromFile(listbox1.Items[k]);
//otwieramy każdy obraz
zamienObrazNaTablice;
//"dekodujemy"
przepiszTablice;
// przepisujemy do tablicy
jednowymiarowej
for i:= 0 to 122499 do
//pętla realizuje zapisanie tablicy
sgb[] do pliku
begin
Write(Plik, floattostr(sgb[i])+' ');
end;
Writeln(Plik, #13); // przejście do
następanej linii
Writeln(Plik, listBox2.Items[k]);
end;
CloseFile(Plik);
end;

```

Aplikacja jest już prawie kompletna brakuje tylko fragmentu kodu odpowiadającego za tworzenie oraz uczenie sieci całość opisana jest poniżej:  
 Tworzymy oraz uczymy sieć neuronową wykorzystując plik utworzony wcześniej:

```

var
mse, dost: single;
begin
fn.Build; // buduj sieć
fn.TrainOnFile(nazwapliku+'grupaMaterialow.data', 100, 0.0000001);
//trenuj sieć
mse:=fn.MSE;

lblMse.Caption:=Format('%%.16f', [mse]);
// wyświetl błąd w odpowiedzi sieci
end;

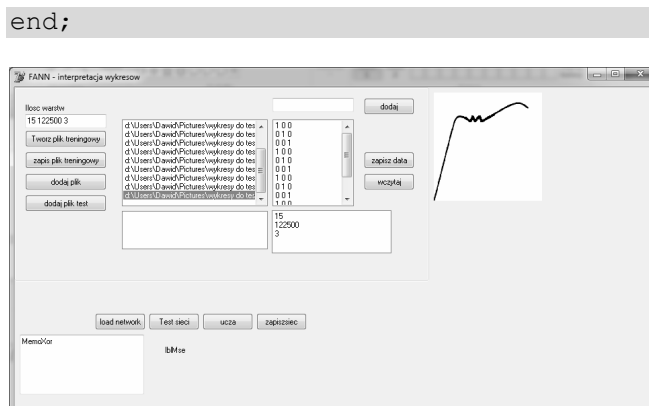
```

Kiedy wynik uczeni sieci jest optymalny zapisujemy sieć do pliku:

```

begin
fn.SaveToFile(nazwapliku+'FanngrupaMaterialow.dat')

```



Rysunek. 5.1 Aplikacja „FANN - interpretacja wykresów”.

### Aplikacja „FANN-materialy”

Aplikacja ta tworzy trzy różne sieci neuronowe jednak opiszę przykład tworzenie pierwsze sieci ponieważ reszta odbywa się analogicznie. W przypadku tego programu danymi uczącymi są liczby odczytane z wykresu statycznej próby rozciągania dlatego nie zachodzi potrzeba „dekodowania” danych do postaci „rozumianej” przez sieć. Pamiętajmy, że komponent FANN 2.0 wykorzystuje inne zmienne środowiskowe niż system Windows z Polską lokalizacją, dlatego należy w opcjach regionu systemu symbol dziesiętny zmienić z przecinka na kropkę. Unikniemy w ten sposób problemów z kompilacją i działaniem programu.  
 Pierwszą czynnością jest wczytanie danych uczących do komponentu StrinGrid realizuje tą czynność procedura Zpliku2StringGrid:

```

Procedure Zpliku2StringGrid(StringGrid:
TStringGrid; FileName: String);
var
File: TextFile;
Tile, i, j: Integer;
Tdane: string;
begin
AssignFile(File, FileName);
Reset(File);
Readln(File, Tile);
StringGrid.ColCount:= Tile;
Readln(File, Tile);
StringGrid.RowCount:= Tile;
for i:=0 to StringGrid.ColCount-1 do
for j:=0 to StringGrid.RowCount-1 do
begin
Readln(File, Tdane);

```

```
StringGrid.Cells[i,j]:= Tdane;  
end;  
CloseFile(File);  
end;
```

poniżej procedura zapisująca dane do pliku:

```
procedure StringGrid2plik(StringGrid:  
TStringGrid;FileName: String);  
var  
File: TextFile;  
i, j: Integer;  
begin  
AssignFile(File, FileName);  
Rewrite(File);  
Writeln(File, StringGrid.ColCount);  
Writeln(File, StringGrid.RowCount);  
for i:=0 to StringGrid.ColCount-1 do  
for j:=0 to StringGrid.RowCount-1 do  
Writeln(File,  
StringGrid.Cells[i,j]);  
CloseFile(File);  
end;
```

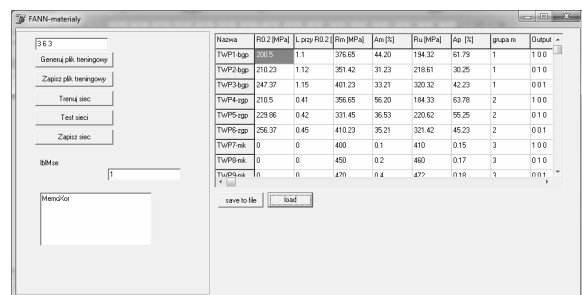
Kolejnym korkiem jest wygenerowanie pliku uczącego zawierające wszystkie dane uczące. Pierwszym krokiem jest stworzenie pliku na dysku z informacjami o ilości danych uczących oraz ilością wejść i wyjść sieci. Realizowane jest to tak samo jak w poprzedniej aplikacji dlatego nie będę już tego opisywał, różnica pojawia się podczas zapisu danych uczących do wcześniej utworzonego pliku:

```
var  
i, j, k: Integer;  
Plik: TextFile;  
begin  
AssignFile(Plik,  
nazwapliku+'material1.data');  
Append(Plik);  
for k:=1 to 3 do  
begin  
for i:= 1 to 6 do  
begin  
Write(Plik,stringgrid1.Cells[i,k]+' ');  
end;  
Writeln(Plik,#13);  
  
Writeln(Plik,stringgrid1.Cells[8,k]);  
end;  
CloseFile(Plik);
```

```
end;
```

Tworzenie oraz uczenie sieci neuronowej wygląda tak samo jak w przypadku wcześniejszej aplikacji jednak dla uniwersalności programu zapis odbywa się za pomocą komponentu Savedialog i wygląda następująco:

```
begin  
if savedialog1.Execute then  
Fanmaterialy.SaveToFile(savedialog1.FileName);  
end;
```



Rysunek. 5.2 Aplikacja „FANN-materialy”.

### Aplikacja „FANN=>identification of the material”

Zajmiemy się teraz sednem tego artykułu, czyli aplikacją wykorzystującą wcześniej stworzone i wytrenowane sztucznie sieci neuronowe do rozpoznawania niektórych metali na podstawie wykresu statycznej próby rozciągania. Aplikacja wykorzystuje fragmenty kodów z wcześniej omówionych programów z tego względu poniżej opisane zostaną tylko nowe części kodu.

Najpierw tworzymy i wczytujemy nauczoną sieć:

```
begin  
fn.Build;  
fn.LoadFromFile(nazwapliku+'FanngrupaMaterialow.dat');  
end;
```

rozpoczynamy interpretacji kształtu wykresu:

```
var  
wejscie: array [0..122499] of single;  
output: array [0..2] of single;  
i,m:integer;  
begin  
result.Clear; //czyścimy memo
```

```
result.Lines.Add('Result of the
identification chart: '); // wstawiamy
tekst
for m:=0 to 122499 do
begin
wejscie[m]:=sgb[m]; //przepisujemy
tablice zawierającą 0 i 1 po dekodowaniu
obrazu
end;
fn.Run(wejscie,output); // uruchamiamy
nauczoną sieć
for i:=0 to 2 do
begin
result.Lines.Add(Format('%0.9f',[Output[i]
])); // wyświetlamy wynik
Identify2.Enabled:=true;
end;
// poniżej lista warunków po spełnieniu
jednego z nich zostaje uruchomiona
odpowiednia
// sieć neuronowa oraz zmienna
rodzajMaterialu i kategoria.
if (Output[0]>= 0.8) and
(Output[0]>Output[1])and
(Output[0]>Output[2]) then
begin
fn2.Build;
rodzajMaterialu:='bez wyraznej granicy
plastycznosci';
fn2.LoadFromFile(nazwapliku+'FannMateria
ly1.dat');
kategoria:='1';
end;
if (Output[1]>= 0.8) and
(Output[1]>Output[0])and
(Output[1]>Output[2]) then
begin
fn2.Build;
rodzajMaterialu:='kruchych';
fn2.LoadFromFile(nazwapliku+'FannMateria
ly3.dat');
kategoria:='2';
end;
if (Output[2]>= 0.8) and
(Output[2]>Output[0])and
(Output[2]>Output[1]) then
begin
fn2.Build;
rodzajMaterialu:='z wyrazna granica
plastycznosci';
```

```
fn2.LoadFromFile(nazwapliku+'FannMateria
ly2.dat');
kategoria:='3';
end;
end;
```

Po zidentyfikowaniu wykresu i wybraniu odpowiedniej sieci rozpoznającej materiały następuje interpretacja danych odczytanych z wykresu:

```
var
wejscie: array [0..5] of single;
output: array [0..2] of single;

i,m:integer;
begin

result.Clear;
result.Lines.Add('Result of the
identification chart: ');
for m:=0 to 5 do
begin
wejscie[m]:=strtofloat(stringgrid1.Cells
[m+1,ComboBox1.ItemIndex+1]);
end;
fn2.Run(wejscie,output);
for i:=0 to 2 do
begin
result.Lines.Add(Format('%0.9f',[Output[i]
]));
end;
memo1.Clear;
if (Output[0]>= 0.8) and
(Output[0]>Output[1])and
(Output[0]>Output[2]) then
begin
Material:='1';
end;
if (Output[1]>= 0.8) and
(Output[1]>Output[0])and
(Output[1]>Output[2]) then
begin
Material:='2';
end;
if (Output[2]>= 0.8) and
(Output[2]>Output[0])and
(Output[2]>Output[1]) then
begin
Material:='3';
end;
```



```
end;
```

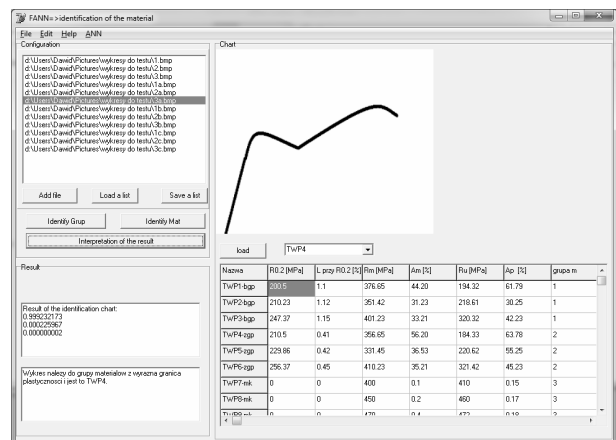
Program posiada już wszystkie dane potrzebne do pełnej rozpoznania materiału na podstawie wykresy statycznej próby rozciągania i realizuje to w następujący sposób:

```
var
nazwaMaterialu:string;
begin
memol.Clear;
if kategoria='1' then
begin
if material='1' then
begin
nazwaMaterialu:='TWP1';
end;
if material='2' then
begin
nazwaMaterialu:='TWP2';
end;
if material='3' then
begin
nazwaMaterialu:='TWP3';
end;
end;
```

```
if kategoria='3' then
begin
if material='1' then
begin
nazwaMaterialu:='TWP4';
end;
if material='2' then
begin
nazwaMaterialu:='TWP5';
end;
if material='3' then
begin
nazwaMaterialu:='TWP6';
end;
end;
```

```
if kategoria='2' then
begin
if material='1' then
begin
nazwaMaterialu:='TWP7';
end;
if material='2' then
begin
```

```
nazwaMaterialu:='TWP8';
end;
if material='3' then
begin
nazwaMaterialu:='TWP9';
end;
end;
memol.Lines.Add('Wykres nalezy do grupy
materialow '+rodzajMaterialu+' i jest to
'+nazwaMaterialu+'.');
end;
```

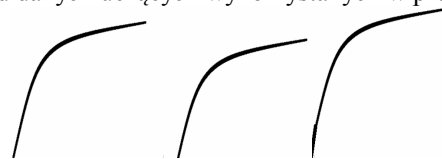


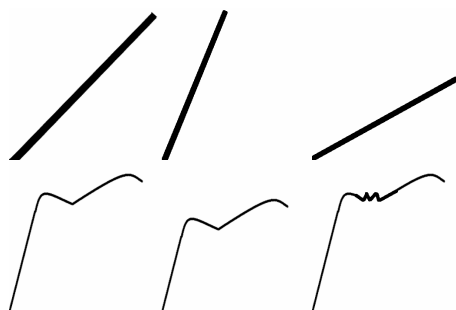
Rysunek. 5.3 Aplikacja „FANN=>identification of the material”

## 6. DANE UCZĄCE

Prawidłowe działanie sztucznej sieci neuronowej uzależnione jest danych uczących. Dlatego bardzo ważnym elementem jest odpowiedni dobór danych uczących oraz strategii uczenia. Ze względu na ilość danych uczących wykorzystanych podczas uczenia sieci przedstawione będzie tylko kilka wybranych przykładów danych uczących. Na rysunku 6.1 prezentowane są przykładowe obrazy przedstawiające kształt wykresy statycznej próby rozciągania metali natomiast na rysunku 6.2 widoczne są przykładowe liczby odczytane z wykresu.

Przykład danych uczących wykorzystanych w programie:





**Rysunek. 6.1** Przykład wykresów - dane uczące zastosowane podczas uczenia sieci.

Nazwa	R0.2 [MPa]	L przy R0.2 [%]	Rm [MPa]	Am [%]	Rlu [MPa]	Ap [%]	Output
TWP1	200.5	0.33	376.65	44.20	194.32	61.79	1 0 0
TWP2	49.85	0.4	321.42	22.23	218.61	29.25	1 0 0
TWP3	247.37	1.1	401.23	33.21	320.32	42.23	1 0 0
TWP4	200.5	0.33	376.65	44.20	194.32	61.79	0 1 0
TWP5	49.85	0.4	321.42	22.23	218.61	29.25	0 1 0
TWP6	247.37	1.1	401.23	33.21	320.32	42.23	0 1 0
TWP7	0	0	400	0.1	410	0.15	0 0 1
TWP8	0	0	450	0.2	460	0.17	0 0 1
TWP9	0	0	470	0.4	472	0.18	0 0 1

**Rysunek. 6.2** Przykład danych odczytanych z wykresu - dane uczące zastosowane podczas uczenia sieci.

## Algorytm uczenia.

Dobór metody uczenia sieci neuronowej jest ważnym elementem, który ma istotny wpływ na działanie sieci. Wybrana metoda została dobrana doświadczalnie i jest jedną z wielu dostępnych w bibliotece FANN. Metoda QUICKPROP powoduje znaczne przyspieszenie procesu uczenia jednocześnie zabezpiecza on przed „wpadnięciem” w płytkie minimum lokalne.

## 7. PODSUMOWANIE

Taka budowa programu daje nam możliwość inteligentnego rozpoznawania metalu na podstawie wykresy statycznej próby rozciągania. Skuteczność programu zależy od ilości danych uczących jednak istnieje możliwość dalszego uczenia sieci co zwiększa możliwości tego narzędzia. Jak widać zastosowanie kilku sieci pozwala porównać program do działania bardzo prostego modelu sztucznego mózgu ponieważ każda z sieci realizuje oddzielne zadanie, jednak wynik końcowy zależy od kolektywnego działania wszystkich sieci.

Jak widać sztuczne sieci neuronowe można z powodzeniem wykorzystać do klasyfikacji danych eksperymentalnych.

## Literatura

1. J. Żurada, M. Barski, W. Jędruch: Sztuczne sieci neuronowe. Podstawy teorii i zastosowania, Wydawnictwo PWN, Warszawa 1996.
2. A. Bochenek, M. Reicher: Anatomia Człowieka, tom IV (Układ nerwowy ośrodkowy), PZWL, Warszawa 2000.
3. J. Korbicz, A. Obuchowicz, D. Uciński: Sztuczne sieci neuronowe - Podstawy i zastosowania, Akademicka Oficyna Wydawnicza PLJ, Warszawa 1994.
4. T. Master: Sieci neuronowe w praktyce, Wydawnictwa Naukowo-Techniczne, Warszawa 1996.
5. PN – EN 10002 - 1:1998. Próba statyczna rozciągania metali.
6. Opis biblioteki FANN 2.0 [online]. [dostęp: 25 marzec 2011].  
Dostępny w Internecie <http://leenissen.dk/fann/wp/>
7. Uczenie sieci neuronowych [online]. [dostęp: 25 kwiecień 2011]. Dostępny w Internecie [http://kopernik.matfiz.polsl.pl/www/Marcin.Wozniak/inne%5Csieci\\_neuronowe/uczenie](http://kopernik.matfiz.polsl.pl/www/Marcin.Wozniak/inne%5Csieci_neuronowe/uczenie)