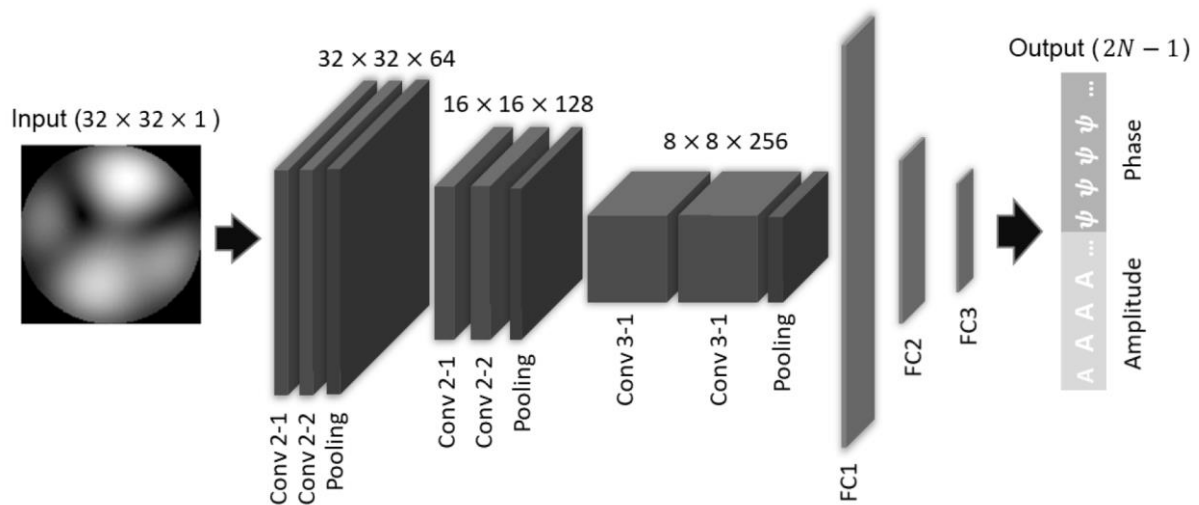


# Protokoll

## Modendekomposition bei einer Multimodefaser



Datum:

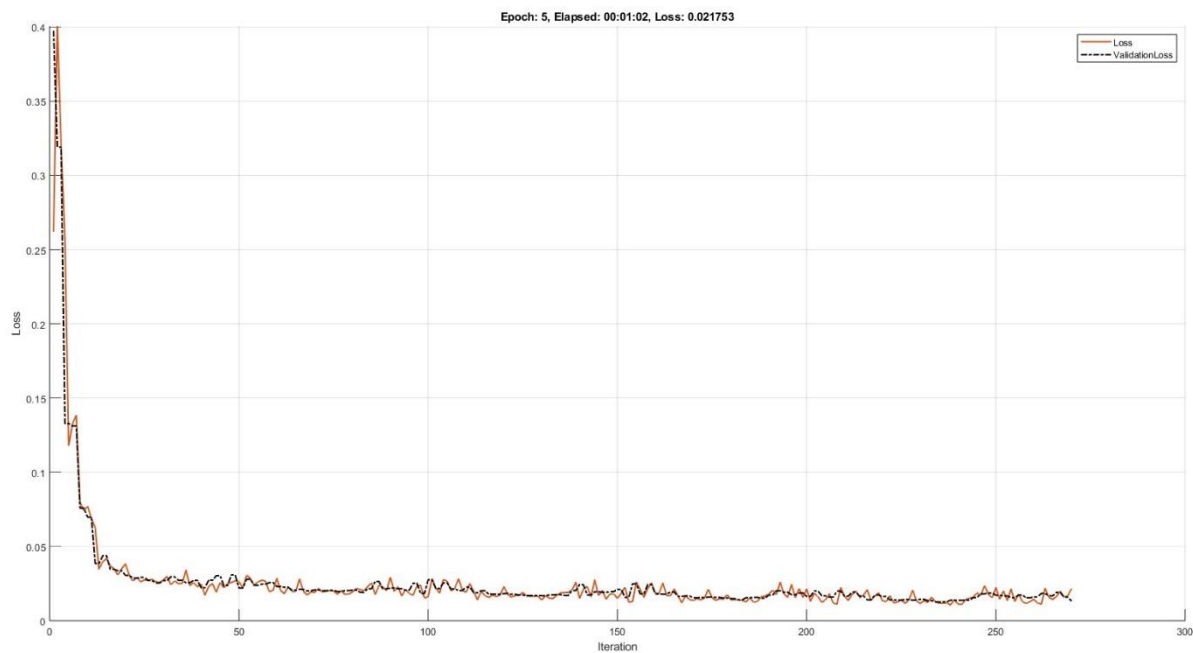
Name	Matrikelnr.	Punkte Protokoll
Hanusch Dustin	4844370	/44

### Aufgabe 1: Training eines MLP Netzes zur Modendekomposition für 3-Moden

Entnehmen Sie der Versuchsanleitung die zur Lösung der Aufgabe notwendigen Schritte 1-3.  
Stellen Sie die Trainingskurve **TrainingLoss** =  $f(\text{iteration})$  dar.

Diagramm (2P):

\*Bonuspunkte (2P): Stellen Sie die Trainingskurven **ValidationLoss** =  $f(\text{iteration})$  in einem einzigen Diagramm dar.



Füllen Sie die Hyperparameter für das Training in der folgenden Tabelle aus.

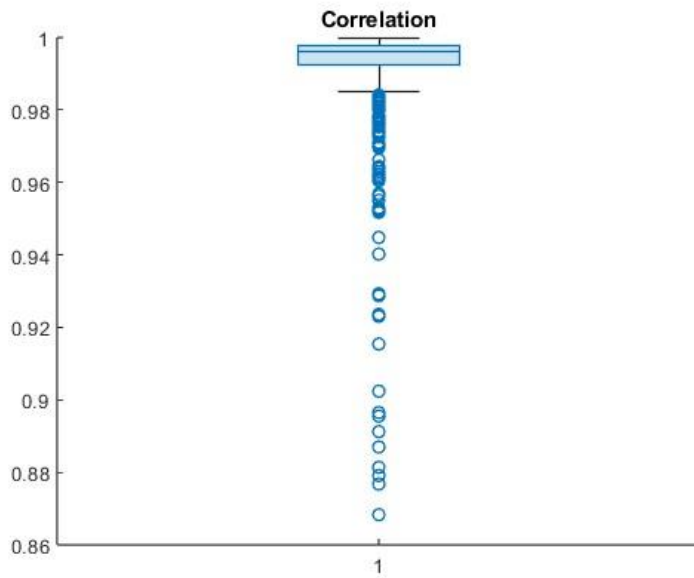
Tabelle (1P):

MLP	Epochs	Learning rate	Mini batch size	Time (min)
3-mode	5	0,001	128	1,03

## Aufgabe 2: Evaluation des trainierten MLP für 3-Moden

Evaluieren Sie die trainierten Netze mithilfe der Testdaten (Schritte 4-5). Visualisieren Sie die Ergebnisse in einem geeigneten Plot mithilfe des Korrelationskoeffizienten ( $\Gamma$ ). Berechnen Sie den Durchschnittswert ( $\bar{\Gamma}$ ) und die Standardabweichung ( $\sigma$ ) der Ergebnisse. Berechnen Sie die relative Abweichung der vorhergesagten Gewichte in Amplitude ( $\rho$ ) und Phase ( $\varphi$ ).

Diagramm (2P):



Füllen Sie das entsprechende Ergebnis in die folgende Tabelle aus.

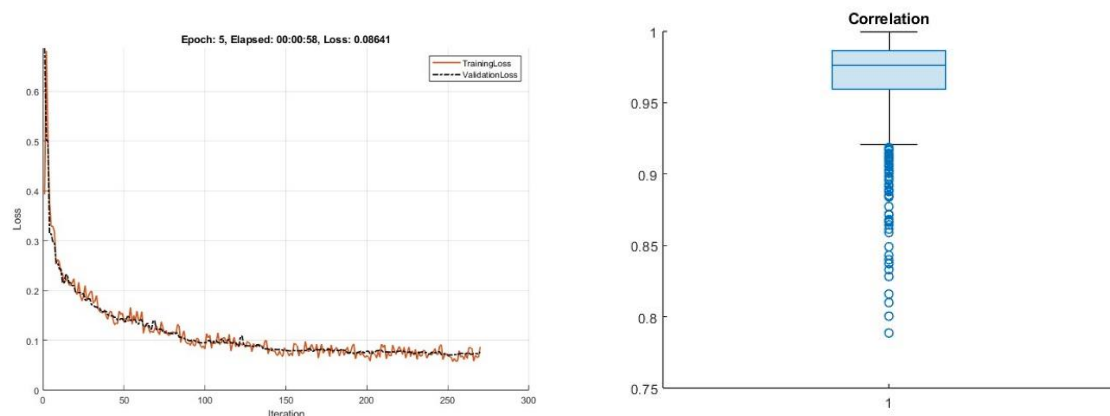
Tabelle (2P):

MLP	$\bar{\Gamma}$	$\sigma$	$\bar{\Delta\rho}$	$\bar{\Delta\varphi}$
3-mode	0,993	0,0124	0,0124	0,0414

### Aufgabe 3: Training und Evaluation eines MLP Netzes zur Modendekomposition für 5-Moden

Stellen Sie die Trainingskurve **TrainingLoss** =  $f(\text{iteration})$  dar und evaluieren Sie die trainierten Netze mithilfe der Testdaten (Schritt 6). Visualisieren Sie die Ergebnisse in einem geeigneten Plot mithilfe des Korrelationskoeffizienten ( $\bar{\Gamma}$ ). Berechnen Sie den Durchschnittswert ( $\bar{\Gamma}$ ) und die Standardabweichung ( $\sigma$ ) der Ergebnisse. Berechnen Sie die relative Abweichung der vorhergesagten Gewichte in Amplitude ( $\rho$ ) und Phase ( $\varphi$ ).

Diagramm (4P):



Füllen Sie die Hyperparameter für das Training in der folgenden Tabelle aus.

Tabelle (1P):

MLP	Epochs	Learning rate	Mini batch size	Time (min)
5-mode	5	0,001	128	0,967

Füllen Sie das entsprechende Ergebnis in die folgende Tabelle aus.

Tabelle (2P):

MLP	$\bar{\Gamma}$	$\sigma$	$\Delta\rho$	$\Delta\varphi$
5-mode	0,967	0,0295	0,0282	0,0705

Beschreibung (1P): (Vergleichen die Ergebnisse von 3- und 5-Modellen)

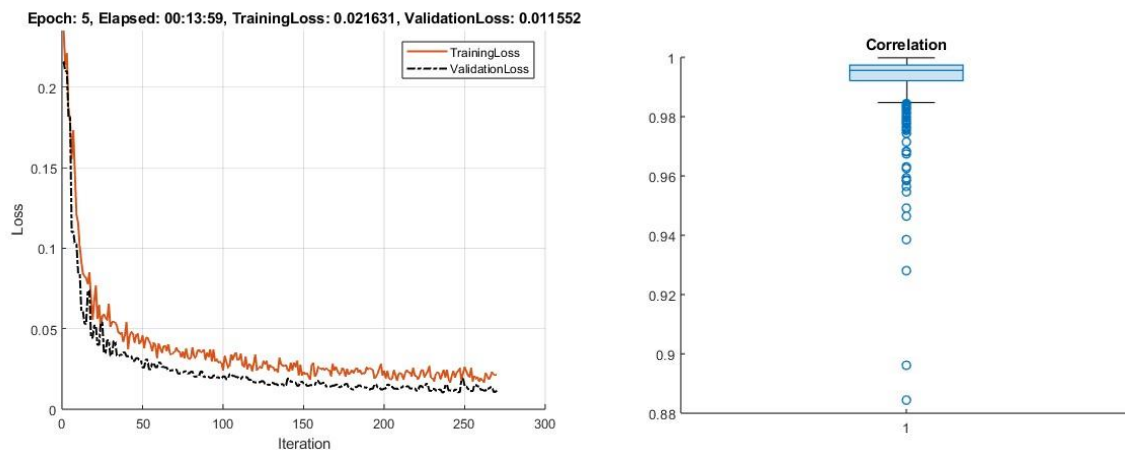
Das Beispiel mit 5 Moden schneidet in allen Gütekriterien schlechter ab. Das ist darauf zurückzuführen, dass bei gleicher Größe des Inputs mehr Daten abgeleitet werden müssen. Bei 5 Moden müssen 9 Werte und bei 3 Moden nur 5 Werte vorhergesagt werden. Dieser generelle Rückschluss ist nur zulässig, da für beide Fälle die identische Netzstruktur mit gleichen Hyperparametern verwendet werden. Die Netze unterscheiden sich nur in der Größe des Output-Layers.

## Aufgabe 4: Training und Evaluation eines VGG Netzes zur Modendekomposition für 3 und 5-Moden

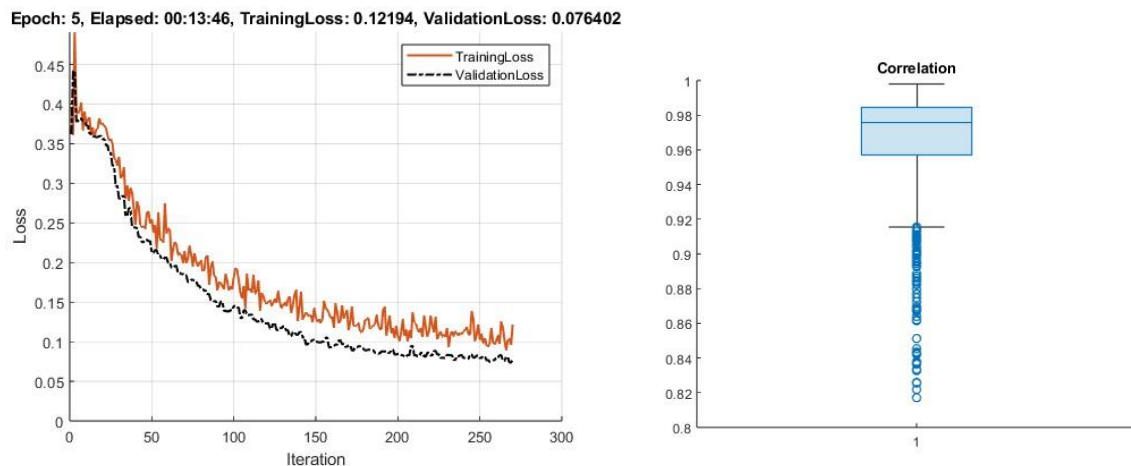
Stellen Sie die Trainingskurve **TrainingLoss** =  $f(\text{iteration})$  dar und evaluieren Sie die trainierten Netze (Schritt 7), wie in Aufgabe 2 und Aufgabe 3. Visualisieren Sie die Ergebnisse in einem geeigneten Plot mithilfe des Korrelationskoeffizienten ( $\Gamma$ ). Berechnen Sie den Durchschnittswert ( $\bar{\Gamma}$ ) und die Standardabweichung ( $\sigma$ ) der Ergebnisse. Berechnen Sie die relative Abweichung der vorhergesagten Gewichte in Amplitude ( $\rho$ ) und Phase ( $\varphi$ ).

Diagramm (8P):

### 3-Moden



### 5 Moden



Füllen Sie die Hyperparameter für das Training in der folgenden Tabelle aus.

Tabelle (1P):

VGG	Epochs	Learning rate	Mini batch size	Time (min)
3-mode	5	0,001	128	13,98
5-mode	5	0,001	128	13,76

Füllen Sie das entsprechende Ergebnis in die folgende Tabelle aus.

Tabelle (4P):

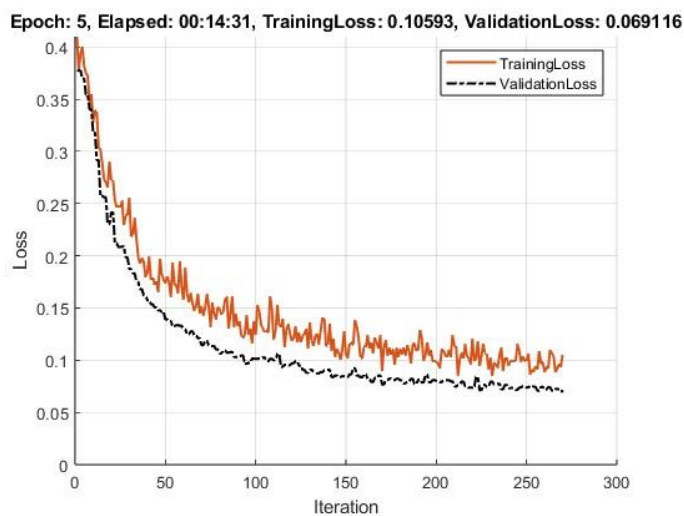
VGG	$\bar{\Gamma}$	$\sigma$	$\overline{\Delta\rho}$	$\overline{\Delta\varphi}$
3-mode	0,994	0,00795	0,0316	0,0154
5-mode	0,966	0,0292	0,0561	0,0611

## Aufgabe 5: Training eines VGG Netzes zur Modendekomposition für 5-Moden durch Transfer Learning

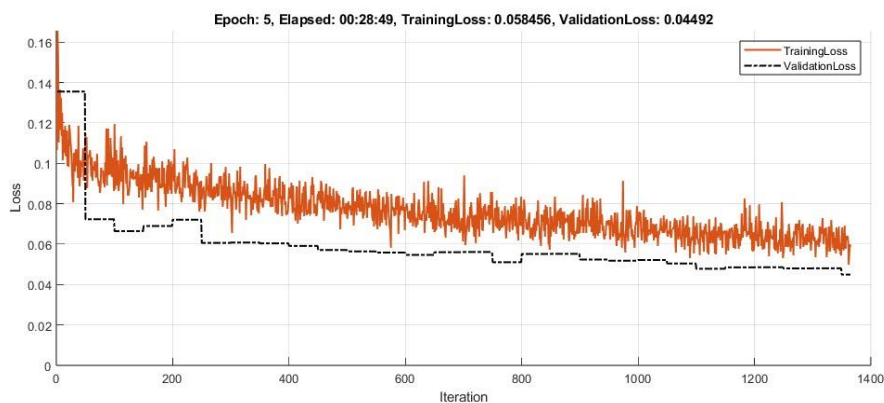
Trainieren Sie ein VGG für 5-Moden mit 10.000 Datenpaaren durch Transfer Learning (Schritt 8). Beachten Sie, dass dieses Training auf dem im Schritt 7 trainierten VGG für 3-Moden basieren soll. Vergleichen Sie die Ergebnisse mit denen des vorherigen Schritts. Erhöhen Sie die Trainingsdaten auf 50.000 Datenpaare für 5-Moden und trainieren Sie das VGG weiter. Verwenden Sie das im Schritt 8 trainierte VGG als Startpunkt. Visualisieren Sie das Ergebnis (Schritt 9).

Diagramm (4P):

### 10.000 Trainingsdaten



### 50.000 Trainingsdaten



Füllen Sie die Hyperparameter für das Training in der folgenden Tabelle aus.

Tabelle (1P):

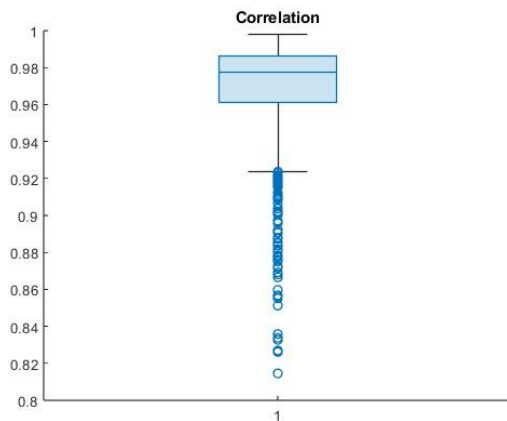
VGG	Training data	Epochs	Learning rate	Mini batch size	Time (min)
5-mode-TL	10000	5	0,001	128	14,51
5-mode-TL	50000	5	0,001	128	28,81

### Aufgabe 6: Evaluation der durch TL trainierten VGG Netze zur Modendekomposition für 5-Moden

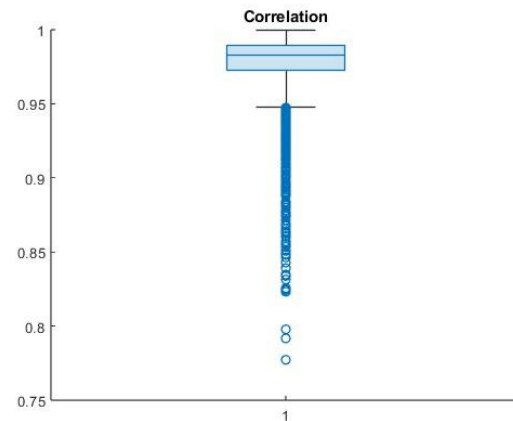
Evaluieren Sie die trainierten Netze (Schritte 8-9). Visualisieren Sie die Ergebnisse in einem geeigneten Plot mithilfe des Korrelationskoeffizienten ( $\Gamma$ ). Berechnen Sie den Durchschnittswert ( $\bar{\Gamma}$ ) und die Standardabweichung ( $\sigma$ ) der Ergebnisse. Berechnen Sie die relative Abweichung der vorhergesagten Gewichte in Amplitude ( $\rho$ ) und Phase ( $\varphi$ ).

Diagramm (4P):

**10.000 Trainingsdaten**



**50.000 Trainingsdaten**



Füllen Sie das entsprechende Ergebnis in die folgende Tabelle aus.

Tabelle (4P):

VGG	Training data	$\bar{\Gamma}$	$\sigma$	$\bar{\Delta\rho}$	$\bar{\Delta\varphi}$
5-mode-TL	10000	0,968	0,0277	0,0486	0,0274
5-mode-TL	50000	0,978	0,01914	0,0262	0,0274



## Aufgabe 7: Diskussion (3P)

Leiten Sie aus den Trainingsergebnissen Zusammenhänge zwischen den untersuchten Anzahl der Moden und Anzahl der Trainingsdaten, sowie die Struktur des neuronalen Netzes ab. Formulieren Sie hierfür eine kurze Diskussion (Stichpunkte erlaubt):

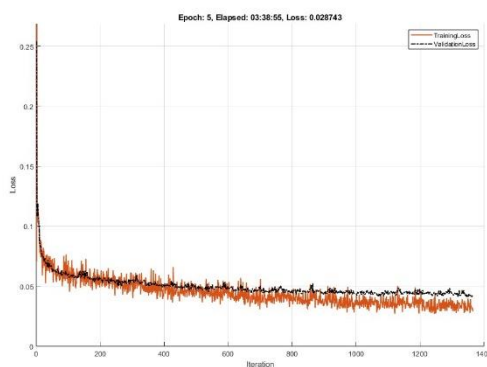
Zum Zweck der Vergleichbarkeit wurden alle Netze bei gleichen Hyperparametern trainiert.

### 1. Trainingszeit

Die Anzahl der Moden hat über alle Netze hinweg keinen großen Einfluss auf die Trainingszeit. Beide MLP's sind extrem schnell in der Berechnung sowie in der Zeit die sie zum Konvergieren brauchen. Bei den VGG's ist die Berechnung deutlich langwieriger, da das Netz wesentlich tiefer ist, dabei ist es unerheblich ob TL angewandt wird oder nicht. Ein wesentlicher Faktor war auch die Validierung. Bei allen Netzen wurde jede zweite Iteration validiert. Das führt dazu dass bei den VGG's eine Zeitdifferenz von circa 200min zwischen 10.000 und 50.000 Daten zustande kommt. Durch Anpassen der Validierungsrate auf 1:50 konnte die Trainingszeit auf 28 min verkürzt werden. Allgemein verlängert eine Erhöhung der Trainingsdaten, sowie eine höhere Anzahl der Validierungsdaten auch die Trainingszeit.

### 2. Overfitting

Bei den ersten Tests zu den VGG ist leichtes Overfitting



aufgetreten. Der Abstand zwischen Validierungsdaten und Trainingsdaten wurde durch Einfügen von zwei Dropout Layern reduziert. Nachteil der Modifikation ist eine langsamere Konvergenz des Netzes. Die MLP's weisen kein Overfitting auf.

### 3. Performance unterschied 5 Moden und 3 Moden

Wie schon in Aufgabe 3 beschrieben gilt dies nur für den Vergleich von Netzen mit gleicher Struktur. So ist es logisch dass alle Netze mit 3 Moden besser performen als die mit 5 Moden. Zu beachten ist dass beim Vergleich von VGG und MLP die Netze nicht identisch

sind. Auch wenn das VGG eine besser angepasste Netzstruktur für die Aufgabe hat, kann es mit 5 Moden und erweitertem Training nicht an die Performance des VGG oder MLP mit 3 Moden heranreichen. Dies kann nur auf die Größere Datenmenge die aus dem Selben Input berechnet werden muss zurückgeführt werden.

Der Vergleich von 5 und 3 Moden im Bereich der Gütekriterien macht im Detail keinen Sinn, da jene mit 5 Moden in nahezu allen Bereichen unterliegen. So werden im Folgenden nur Netze mit gleichen Moden verglichen.

#### 4. Gütekriterien

##### Vergleichstabelle:

Netz	Trainingsdaten	$\bar{r}$	$\sigma$	$\overline{\Delta\rho}$	$\overline{\Delta\varphi}$
MLP 3	10.000	0,993	0,0124	0,0124	0,0414
VGG 3	10.000	0,994	0,0079	0,0316	0,0154
MLP 5	10.000	0,967	0,0295	0,0282	0,0705
VGG 5	10.000	0,966	0,0292	0,0561	0,0611
VGG 5 TL1	10.000	0,968	0,0277	0,0486	0,0274
VGG 5 TL2	50.000	0,978	0,0191	0,0262	0,0274

Allgemein ist zu erkennen das bei 10.000 Trainingsdaten die (MLP 3 vs VGG 3, MLP5 vs VGG5) die Korrelation sehr ähnlich ist. Jedoch können die VGG's bei so wenig daten die Phase und die MLP's die Amplitude besser berechnen. Trotzdem ist das VGG besser da die Standartabweichung geringer ist und sie somit konsistentere Ergebnisse liefern.

Beim TL1 erkennt man das es nicht sonderlich effektiv ist auf ein Netz mit anderer Output-Size aufzubauen, auch wenn sich die relative Phasendifferenz weiter verbessert.

Beim TL2 überholt das VGG dann in allen Bereichen das MLP 5. Daran erkennt man das das VGG besser auf die Aufgabe angepasst ist, da mit mehr Trainingszeit und Trainingsdaten eine bessere Qualität erreicht werden kann. Beim MLP hätte mehr zeit zu keiner signifikanten Besserung geführt, da es bereits nach wenigen Iterationen konvergiert. Im letzten Versuch war der Vorteil von TL deutlich zu sehen. Bereits am beginn ist der Loss fast so gering wie beim TL1 am Ende. Somit wird das bisherige netzt nur noch mit mehr Daten verfeinert, was zur allgemein besten Leistung führt.