

Inhaltsverzeichnis

1 Einleitung	3
2 Grundlagen	5
2.1 Konventionelle Metriken	5
2.2 Perzeptuelle Metriken	7
2.2.1 Modellbasierter Ansatz	7
2.2.2 Signalbasierter Ansatz	8
2.2.3 Hybrider Ansatz	8
2.2.4 Beispiel Structural Similarity Index	9
2.3 Subjektive Qualitätstests	11
2.3.1 Überblick	11
2.3.2 Planung	11
2.3.3 Testmethoden	15
2.3.4 Bewertungsmethoden	20
2.3.5 Auswertungsmethoden	22
2.4 Zusammenfassung	24
3 Konzept	25
3.1 Crowdsourcing	25
3.2 Datenbank	26
3.3 Datenanalyse	27
3.4 Architektur	27
4 Implementierung	29
4.1 Symfony	29
4.2 Controller	31
4.3 Datenbank	33
4.4 Präsentation	35
4.5 Ergänzendes	36
4.6 Datenanalyse	37
4.7 Setup	38
5 Laboreinrichtung	39
5.1 Ausstattung	39
5.2 Messungen	40
6 Experiment	42
6.1 Konfiguration	42
6.2 Durchführung	45
6.3 Auswertung	45
7 Fazit und Ausblick	51
Literatur	53
Anhang	54

Zusammenfassung

Im Rahmen dieser Abschlussarbeit wurde eine Software entwickelt mit der es möglich ist Bild- und Videoqualitätstests durchzuführen. Die Grundlage dieser Implementierung stellen die Referenzen BT.500, P.910 und BT.1788 der International Telecommunication Union dar. Auf Basis dieser Dokumente und weiteren Anforderungen wird ein Konzept für eine Applikation zur webbasierten Durchführung von Assessments entworfen. Eine Ausarbeitung der Richtlinien, die konkrete Implementierung der Software und die Einrichtung eines Versuchslabors stellen den Hauptteil dieser Arbeit dar. Ergänzend dazu werden mehrere Assessments durchgeführt, die zur Validierung der Software beitragen sollen.

1 Einleitung

Motivation

Ein grundlegendes Problem der Bild- und Videokodierung ist die Bewertung von Bildqualität und Wiedergabetreue der kodierten Sequenzen. Ein typischer Anwendungsfall in dem diese Bewertungen dringend benötigt wird ist bspw. der Systemvergleich von unterschiedlichen Kodierverfahren. Häufig können daraus Erkenntnisse abgeleitet werden, die dann zu einer Optimierung der Kodierverfahren beitragen. Andere Anwendungsfälle benötigen die Einschätzung der Bildqualität sogar bereits während des Kodervorgangs. Auf Grundlage der Rate-Distortion-Theorie wird bspw. beim neuen Videostandard H.265 eine kontinuierliche Kostenoptimierung durchgeführt, bei der die benötigten Bitkosten mit der resultierenden Bildqualität abgewogen werden [VS14]. Da diese Bewertung nicht immer durch den eigentlichen Konsumenten vorgenommen werden kann, werden vielfältige Formalismen, sogenannte *Image Quality Metrics* (IQMs), für diese Bewertungen verwendet. Ein vielverwendeter Ansatz ist die Anwendung von mathematischen Fehlermaßen, wie dem *Peak Signal-to-Noise Ratio* oder dem *Mean Squared Error*. Diese Metriken bieten dafür zwar eine Vielzahl an wünschenswerten Eigenschaften, berücksichtigen aber keine spezifischen Besonderheiten des *Human Visual System* (HVS) für die Bewertung, wie unterschiedliches Helligkeits- und Farbempfinden oder Kontrastsensitivität. Jedoch sind besonders diese Eigenschaften für die bereits erreichten Komprimierungsgewinne bei der Kodierung von Bild- und Videodaten mitverantwortlich und sollten daher umbedingt berücksichtigt werden. Um dem menschlichen Sehvermögen daher zunehmend gerecht zu werden, ist den sogenannten PVQMs (*Perceptual Visual Quality Metrics*) eine große Aufmerksamkeit zugekommen. Dies umfasst eine Vielzahl an komplexen Metriken, die auf verschiedenen Erkenntnissen aus Untersuchungen über das HVS aufbauen und dafür seit Jahren entwickelt und erweitert werden.

Kritische Untersuchungen zeigten jedoch wiederholt, dass selbst etablierte und hochkomplexe Qualitätsmetriken keine ausreichende Korrelation mit subjektiven Referenzbewertungen aufweisen [LJ11] [VQE10] [DY09]. Diese Untersuchungen nutzen dabei sogenannte *Image Quality Assessments* (IQAs), bei denen Menschen in einer wohldefinierten Laborumgebung ihre Bewertungen für die Sequenzen abgeben. Dabei unterliegen diese Assessments einer erheblichen Anzahl an Störeinflüssen und Randbedingungen und haben zudem auch finanzielle und organisatorische Grenzen. Dennoch sind diese Bewertungen als Referenz- und Kontrollmöglichkeit unerlässlich und in vielen Fällen nicht sinnvoll durch Metriken ersetzbar.

Ziel der Arbeit

Auch im Rahmen der Forschungsarbeiten am Fraunhofer Heinrich Hertz Institut entstehen regelmäßig Fragestellungen, deren bestmögliche Beantwortung nur durch Assessment erfolgen kann. Darum soll im Kontext dieser Abschlussarbeit eine Möglichkeit geschaffen werden, die es erlaubt Bild- und Videoqualitätstests durchzuführen und auszuwerten. Ein speziell dafür eingerichteter Raum liefert dahingehend bereits einige Grundlagen. Um diese Infrastruktur zu nutzen, muss das Versuchslabor eingerichtet und eine geeignete Software entwickelt werden. Die Rahmenbedingungen dieser Testumgebung sollen dafür auf Grundlage der ITU-Richtlinien BT.500, P.910 und BT.1788 gestellt werden. Die dort vorgestellten Testverfahren, Auswertungsmechanismen und Rahmenbedingung für solche Assessments, sollen dafür zunächst ausgearbeitet und anschließend in ein geeignetes Softwarekonzept übertragen werden. Die anschließende Umsetzung dieses Konzepts in eine lauffähige Software stellt dabei das Hauptziel dieser Arbeit dar. Ein weiterer und wichtiger Bestandteil dieser Arbeit ist außerdem die anschließende Durchführung eines Assessments mit der Testumgebung, um die Funktionstüchtigkeit unter realen Bedingungen zu prüfen.

Struktur der Arbeit

Zur Bearbeitung der Aufgabenstellung werden einleitend die Grundlagen von Bild- und Videoqualitätstests aus den entsprechenden ITU-Richtlinien erarbeitet. Dafür werden die verschiedenen Test- und Bewertungsverfahren vorgestellt, aber auch die organisatorischen und wissenschaftlichen Komponenten solcher Assessments aufgezeigt. Dieses Wissen dient anschließend der Entwicklung eines entsprechenden Softwarekonzepts. Dabei werden die zentralen Ideen und Designentscheidung des Vorhabens skizziert und eine adäquate Systemarchitektur geplant. Der nächste Abschnitt wird sich dann mit der Implementierung der Software beschäftigen. Hier wird auf die wesentlichen Aspekte des Softwareengineering eingegangen und die grundlegende Mechanik der verwendeten Frameworks und des Datenbanksystems erläutert. Im letzten Teil dieser Arbeit wird mit der entwickelten Anwendung ein experimentelles Assessment durchgeführt. Dafür hat eine Vielzahl an Probanden unter festgelegten Bedingungen diverse Sequenzen bewertet. Eine umfangreiche Analyse der Daten stellt dann den Abschluss dieser Arbeit dar, mit der die Funktions tüchtigkeit der Anwendung verifiziert werden soll.

2 Grundlagen

2.1 Konventionelle Metriken

Die konventionellen und dominanten Metriken in der Signalverarbeitung sind die signalbezogenen Metriken. Sie sind ein oft verwendetes Standardkriterium für die Beurteilung von Signalqualität und Wiedergabetreue für den Vergleich unterschiedlicher Systeme und somit auch bei deren Optimierung [WB09]. Die bekanntesten Metriken sind laut [LJ11]: MAE, MSE, SNR und PSNR. Diese werden in [VA11] wie folgt definiert:

MAE : Mean Absolute Error

$$\text{MAE}(\mathbf{x}, \mathbf{y}) = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N |x(i, j) - y(i, j)|$$

$x(i, j)$ repräsentiert das Referenzbild und $y(i, j)$ die beeinträchtigte Sequenz mit den Pixelpositionen i und j in dem $N \cdot M$ großen Bild.

MSE : Mean Squared Error

$$\text{MSE}(\mathbf{x}, \mathbf{y}) = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (x(i, j) - y(i, j))^2$$

SNR : Signal Noise Ratio

$$\text{SNR} = 10 \log \frac{P_{\text{Signal}}}{P_{\text{Rauschen}}} dB$$

PSNR : Peak Signal-to-noise Ratio

$$\text{PSNR} = 10 \log \frac{\text{MAX}^2}{\text{MSE}} dB$$

MAX = Maximal möglicher Wert zB. 255 bei 8 Bit Farbtiefe. ($2^n - 1$)

Je größer der PSNR ist umso besser ist die Bildqualität, da weniger Verfälschungen vorhanden sind.

Ein großer Vorteil dieser signalbezogenen IQMs ist ihre Einfachheit. Sie sind gewöhnlich parameterfrei und vergleichsweise kostengünstig zu berechnen. Sie bieten zudem einige mathematische/statistische Vorteile und sind daher besonders bei Optimierungsproblemen ein wünschenswertes Maß. So werden bspw. für den MSE Symmetrie, Linearität, Differenzierbarkeit und Konvexität genannt. Darüber hinaus haben sie außerdem eine klare physikalische Bedeutung, was sie sehr anschaulich macht. Aus diesen Gründen sind sie auch so weit verbreitet und anerkannt. [WB09]

Die wesentlichen Schwächen der signalbezogenen Metriken zeigen sich vorwiegend bei der Anwendung auf perzeptuell relevante Signale, wie Audio und Video [WB09]. Dabei fasst [LJ11] diese Nachteile in 4 grundlegende Probleme zusammen:

- 1 Nicht jede Veränderung im Bild kann durch den Betrachter bemerkt werden.
(Farbunterabtastung, verschiedene Frequenzen)
- 2 Nicht jede Region bekommt dieselbe Aufmerksamkeit vom Betrachter.
(Objekte in Bewegung, Fokussierungseigenschaft des HVS)
- 3 Nicht jede Veränderung führt zu einer Verschlechterung des Bildes, sondern kann auch eine Verbesserung des subjektiven Eindrucks bewirken.
(Postprocessing zur Bildverbesserung)
- 4 Gleichstarke Änderungen führen nicht automatisch zu einer gleichstark wahrnehmbaren Veränderung des Bildes. (Nichtlinearität des HVS)

Veranschaulicht werden diese Überlegungen durch folgende Gegenüberstellungen:

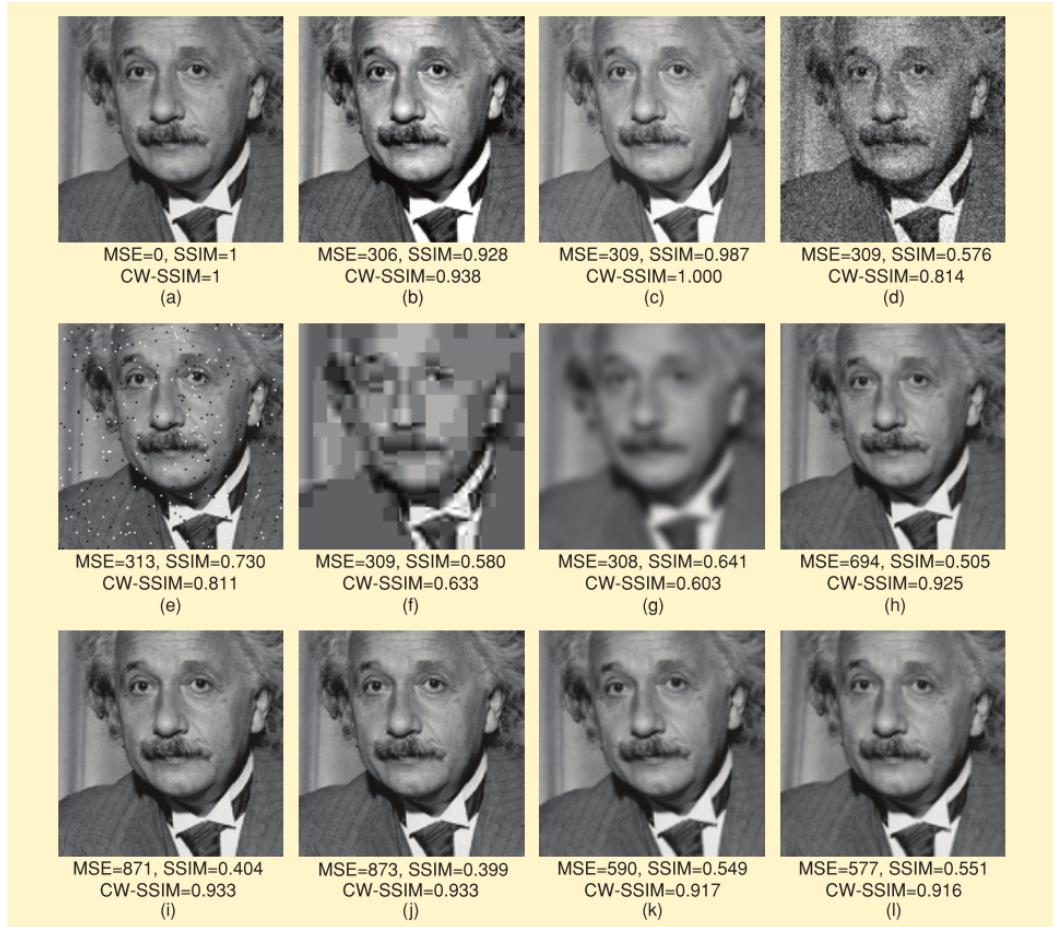


Abbildung 1: Unterschiedlich gestörte Varianten der Originalsequenz (a) [WB09]

Bei der Betrachtung der Varianten (b) bis (g) in Abb. 1 sind für den menschlichen Betrachter deutliche Qualitätsunterschiede zu erkennen, während der MSE einen relativ konstanten Wert von rund 310 beibehält. Sehr schlecht bewertet der MSE hingegen die letzten 4 Sequenzen, obwohl diese nur minimal rotiert ((k) und (l)) bzw. verschoben ((i) und (j)) wurden und sonst mit dem Originalbild identisch sind. Das dieses Verhalten nicht dem natürlichen Qualitätsempfinden entspricht ist klar erkennbar. Die objektive Bewertung unter Berücksichtigung des HVS kann demnach durch signalbezogene Metriken nicht sinnvoll realisiert werden, da sie wesentliche Eigenschaften des menschlichen Sehvermögens nicht berücksichtigen [WB09].

2.2 Perzeptuelle Metriken

Die perzeptuellen Metriken oder auch *Perceptual Visual Quality Metrics* (PVQMs) sind das Ergebnis des Versuchs die Schwächen der signalabhängigen, konventionellen Metriken durch neue Ansätze auszugleichen. Dafür wurden neue Konzepte entwickelt, die Rücksicht auf die Eigenschaften des HVS nehmen sollen [LJ11]. Eine Unterscheidung verschiedener PVQMs findet dabei vor allem an Hand ihrer Modellierung in modell- und signalbasierte Verfahren statt. Eine weitere Aufgliederung teilt diese Verfahren in sogenannte *Single-ended-* und *Double-ended-Metriken* ein. *Single-ended-Metriken* beschreiben Verfahren, die zur Bewertung lediglich das beeinträchtigte Bild verwendeten und daher auch als *No-Reference-Metriken* bezeichnet werden. *Double-ended-Metriken* hingegen benötigen neben dem Testbild auch das Referenzbild für die Bewertung. Diese Metriken werden außerdem noch weiter in *Full-reference-* oder *Reduced-reference-Metriken* unterschieden. Eine *Full-reference-Metrik* verwendet das gesamte Referenzbild für die Beurteilung der Bildqualität, während eine *Reduced-reference-Metrik* nur einen Ausschnitt des Referenzbildes nutzt. Einige der populärsten PVQMs sind: *Structural SIMilarity index* (SSIM), *Multi-resolution Singular Value Decomposition* (MSVD), *Visual Information Fidelity* (VIF), *Information Fidelity Criterion* (IFC) und *Visual Signal-to-Noise Ratio* (VSNR). [LJ11]

2.2.1 Modellbasierter Ansatz

Die modellbasierten (*model-based* oder auch *vision-based*) Metriken werden anhand von systematischer Modellierung entwickelt. Diese Modelle werden auf Grundlage der Erkenntnisse über das menschliche Sehvermögen entwickelt. Die Erforschung dieser Grundlagen stellt eine interdisziplinäre Aufgabe dar, weil hier neurophysiologische Effekte und eine Vielzahl an psychologischen Mechanismen berücksichtigt werden müssen, die bis heute nicht alle vollständig verstanden worden sind [LJ11]. Dabei ist aber dieses Wissen über das HVS ein wichtiger Teil bei der Entwicklung von Metriken, die unter menschlichen Gesichtspunkten eine Beurteilung der Qualität vornehmen sollen. Darüber hinaus ist dieses Verständnis, nicht nur in der Vergangenheit, ein wichtiger Punkt für die Optimierungsgewinne in der Bild- und Videokodierung gewesen und wird es auch weiterhin sein [WB09]. Einige bekannte Eigenschaften des menschlichen Auges sind: Mehr Zapfen (*cones*) als Stäbchen (*rods*), was die Ursache für ein besseres Helligkeits- als Farbempfinden ist; die begrenzte Anzahl der Stäbchen, wodurch das Auge wie einen Tiefpass unempfindlicher für hohe Frequenzen ist; diverse Maskierungseffekte und eine stärkere Empfindlichkeit für Objekte, die in Bewegung sind [Win13].

Erste *Single-channel-IQMs* gingen daher zunächst von der Annahme aus, bei dem HVS handle es sich lediglich um einen einfachen Raumfilter mit einer charakteristischen Kontrastfunktion (CFS). Bessere aber aufwendigere Verfahren wurden mit den Jahren, durch Nutzung des Referenzbildes, Signalzerlegung und der Auswertung von Kontrast und Maskierungseffekten, entwickelt. [LJ11]

Die steigende Komplexität der modellbasierten Metriken und das weiterhin lückenhafte Verständnis über das HVS, führte zu keinem großen Erfolg [LJ11]. Nach Meinung von [WB09] gibt es darüber hinaus zunehmend Zweifel an der Vermutung, dass die präzise Simulation aller Komponenten des menschlichen Wahrnehmungsvermögens zu einer korrekten Prädiktion von Bildqualität führen wird [WB09].

2.2.2 Signalbasierter Ansatz

Bei signalbasierten (*signal-driven*) Metriken wird versucht das bestehende Wissen über die Signalgewinnung und Analyse zu nutzen, um neue Ansätze daraus abzuleiten. Sie bauen meist auf Erkennungsverfahren für bekannte Eigenschaften von kodierten Bildern auf und nutzen die enthaltenen strukturellen Informationen. Dafür werden die Sequenzen gezielt nach verschiedenen Phänomenen wie Artefakten, Blockbildung, Unschärfe oder Helligkeits- und Kontrastverzerrungen untersucht. Je nach Ausprägungsstärke, Vorkommen und Relevanz ergeben diese Merkmale dann das Resultat der Metrik. [LJ11]

Ein beispielhaftes Verfahren für die Erkennung von Blockbildung (*Blockiness*), dass durch die blockbasierte DCT-Kodierung besonders bei niedrigen Bitraten entsteht, ist folgendes: Bei einem $H \times W$ großem Beispielbild können die Grenzen der $N \times N$ großen Blöcke durch die nachfolgende Funktionen berechnet werden. Die Blockbildung selbst wird dafür an den einzelnen Stellen durch eine Intensitätsfunktion I ausgewertet. [LJ11]

Für horizontale Blockbildung:

$$M_h = \left[\sum_{k=1}^{H/N-1} \sum_{x=0}^{W-1} (I(x, k \cdot N - 1) - I(x, k \cdot N))^2 \right]^{1/2}$$

Für vertikale Blockbildung

$$M_v = \left[\sum_{l=1}^{W/N-1} \sum_{y=0}^{H-1} (I(l \cdot N - 1, y) - I(l \cdot N, y))^2 \right]^{1/2}$$

Dies ist auch ein gutes Beispiel dafür, dass diese Methoden nicht einfach auf jeden Anwendungsfall übertragbar sind. Da bspw. die vorhandene Annahme von gleich großen Blöcken, bei der neuen Partitionierungstechnik von H.265 keine korrekten Ergebnisse liefern würde [VS14]. Daher muss auch hier eine Vielzahl von unterschiedlichen Implementierungen und alternativen Verfahren entwickelt werden, die anderen Anwendungsfällen genügen [LJ11].

2.2.3 Hybrider Ansatz

Zunehmend entstehen aber auch Modelle, die beide Ansätze kombinieren. In [WM08] wird unter anderem von einem Ansatz berichtet, der zwischen modellbasiertem und signalbasiertem Ansatz wechselt. Der signalbasierte Ansatz wird für die Bewertung von *Blockiness* genutzt, während der modellbasierte Ansatz dann in stark betroffenen Gebieten weiterverwendet wird. [LJ11]

2.2.4 Beispiel Structural Similarity Index

Für einen tieferen Einblick in die Funktionsweise und Komplexität einer IQM, kann es hilfreich sein, sich eine Metrik im Detail anzusehen. Der *Structural SIMilarity (SSIM) index* ist eine verbreitete Full-reference-Metrik und ein damit ein geeigneter Stellvertreter für die PVQMs. Diese Metrik vergleicht für die Bewertung der Bildqualität die strukturelle Gleichheit des ungestörten Referenzbildes mit dem beeinträchtigten Testbild und liefert als Ergebnis einen Wert im Intervall $-1 < S(x, y) \leq 1$. Dabei bedeutet der Wert 1 eine ideale Übereinstimmung mit dem Referenzbild und negative Werte eine Inversion.

Die grundlegende Idee des SSIM wurde aus der Erkenntnis abgeleitet, dass das HVS sehr gut strukturelle Informationen extrahieren kann. Diese Tatsache folgt aus der Beobachtung das strukturelle und nichtstrukturelle Beeinträchtigungen eine unterschiedliche Wirkung auf die Gesamtqualität haben. Die nichtstrukturellen Beeinträchtigungen, wie Helligkeits- oder Kontrastabweichungen, bewirken dabei keine wirkliche Veränderung an den Objekten, die im Bild zu erkennen sind. Anders ist dies bei strukturellen Beeinträchtigungen, wie Rauschen oder Blockbildung. Diese bewirken, dass Bildobjekte schlechter zu erkennen sind und verfremdet werden. [WB09]

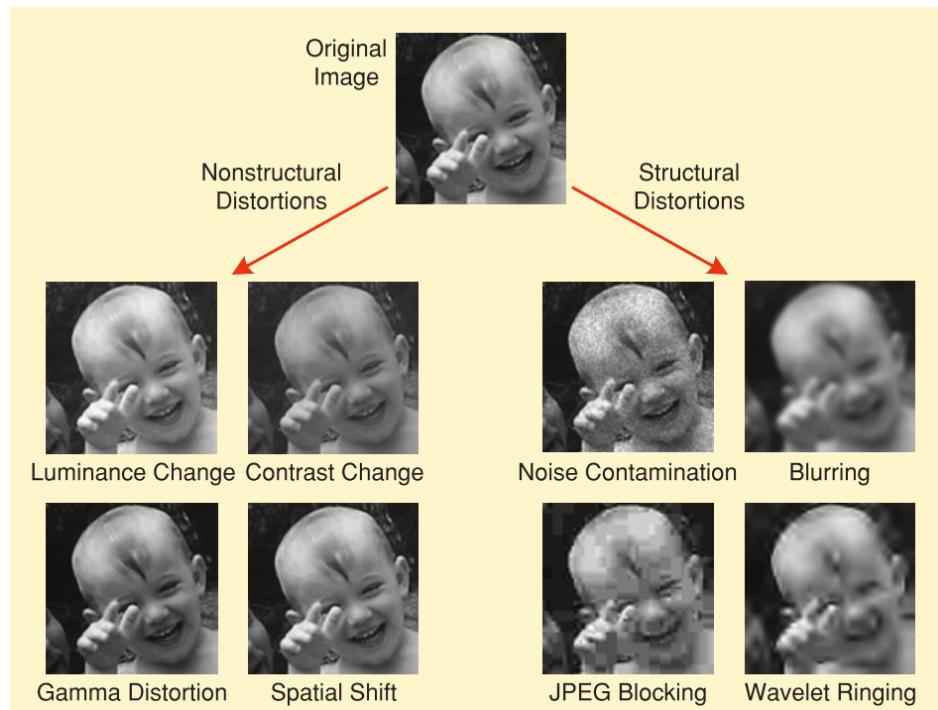


Abbildung 2: Beispiele für strukturelle und nichtstrukturelle Beeinträchtigungen [WB09]

Um diese Funktionalität in ein berechenbares Modell zu übertragen, wird die Gleichheit von Helligkeit $l(x,y)$, Kontrast $c(x,y)$ und struktureller Übereinstimmung $s(x,y)$ der beiden Bildsignale x und y wie folgt berechnet:

$$SSIM(x, y) = l(x, y) \cdot c(x, y) \cdot s(x, y) = \left(\frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \right) \cdot \left(\frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \right) \cdot \left(\frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \right)$$

Dabei ist μ der Mittelwert, σ die Standardabweichung, und σ_{xy} die Kovarianz (Kreuzkorrelation ohne Mittelwert). Die Konstanten (C_1, C_2) sollen kleine positive Werte sein und dienen lediglich der numerischen Stabilität des Verfahrens. [WB09] [WaBSS04]

Laut [WB09] wird typischerweise mit einem 8x8 Pixel großen Bildausschnitt gerechnet, der pixelweise über das Bild verschoben wird. Begründet wird dieses Vorgehen damit, dass statistische Eigenschaften von Bildern oder Bildverzerrungen typischerweise nicht örtlich gebunden sind. Außerdem entspricht dies auch dem Verhalten des menschlichen Auges, das dazu neigt einzelne Bildbereiche stärker zu fokussieren. Es gibt allerdings eine Vielzahl an Modifikation des SSIM, um bspw. eine stärkere Gewichtung der strukturellen gegenüber den nichtstrukturellen Eigenschaften vorzunehmen. Im Vergleich mit dem MSE schneidet der SSIM in vielen Fällen bei der Erkennung von Bildstörungen besser ab. [WB09]

Erste Schwächen zeigt der SSIM allerdings schon bei seiner Empfindlichkeit gegenüber relativen Verschiebungen, Skalierungen und Rotationen von Bildern, wie es auch schon beim MSE zu beobachten war. Dieses Verhalten kann dafür wieder sehr gut Anhand der letzten vier Beispiele in Abb. 1 nachvollzogen werden, die dafür leicht skaliert und verschoben wurden. Dieser Sachverhalt steht deutlich im Widerspruch mit der ursprünglichen Idee von struktureller Gleichheit. Eine Lösung dieses Problems liefert dafür bspw. der CompelexWavelet-SSIM (CW-SSIM). Hierbei wird das Bildsignal erst in der komplexen Spektral- bzw. Waveletdarstellung untersucht, wodurch bessere Ergebnisse erzielt werden können [WB09].

Letztlich konnte mit Hilfe einer empirischen Studie und einer formaler Analyse außerdem bereits eine Beziehung zwischen der Bewertung durch MSE und SSIM nachgewiesen werden [DY09]. Dafür wurde zunächst ein statistischer Vergleich diverser subjektiver und objektiver Bewertungen vorgenommen. Daraus konnte ein mathematisches Modell abgeleitet werden, mit dem anschließend eine formale Verbindung zwischen SSIM und dem MSE aufgezeigt wird. Da dies gelingt, ist davon auszugehen, dass der SSIM nicht weiter als ein verbesserter MSE ist. [DY09]

2.3 Subjektive Qualitätstests

2.3.1 Überblick

Den subjektiven Tests kommt eine Vielzahl an wichtigen Aufgaben im Bereich der Qualitätsbewertung von Bild- und Videodaten zu. Dabei ist es keineswegs eine einfache Aufgabe einen subjektiven Qualitätstest durchzuführen, da es typischerweise eine erhebliche Anzahl an Störeinflüssen und Randbedingungen gibt, die zu berücksichtigen sind. So können unter anderem die physikalische Umgebung, ein technischer Parameter oder die emotionale Verfassung der Testperson einen dramatischen Einfluss auf die Messgrösse haben und so Resultate verfälschen. Außerdem unterliegen diese Experimente zudem weiteren finanziellen und organisatorischen Grenzen, die sie zusätzlich erschweren. Um trotzdem Untersuchungen zu ermöglichen, werden daher vertrauenswürdige, nachvollziehbare und vergleichbare Experimente benötigt. Eine umfangreiche Referenz für die Durchführung solcher Tests wird dabei von der *International Telecommunication Union (ITU)* zur Verfügung gestellt.

Die ITU Richtlinien P.910 [IT08] und BT.500 [IT12] stellen dafür verschiedene Methoden vor, um Bild- und Videoqualitätstests für Multimedia (P.910) und TV-Anwendungen (BT.500) durchzuführen. Beide gehen auf Testdesign und Testmethoden ein, nennen Anforderungen für das Testmaterial und stellen Möglichkeiten zur Datenanalyse vor. Letztlich wird auch auf häufige Fehler und ungeklärte Probleme eingegangen, die im Kontext dieser Assessments häufig anzutreffen sind. Zunächst werden daher im Folgenden die organisatorischen und technischen Grundlagen dieser Assessments beschrieben. Anschließend folgt eine Beschreibung von *Single Stimulus* (SS), *Double Stimulus* (DS) und *Pair Comparison* (PC) Verfahren aus [IT08] und [IT12]. Des weiteren wird auch das alternative SAMVIQ-Verfahren aus ITU Richtlinie BT.1788 [IT07a] vorgestellt. Die verschiedenen Bewertungsmöglichkeiten und Auswertungsmethoden stellen dann das Ende dieser Ausarbeitung dar.

2.3.2 Planung

Die Gestaltung des Ablaufes eines Assessments hat entscheidende Auswirkungen auf Aufwand, Resultate und Kosten des Tests. Dabei erfordern unterschiedliche Zielsetzungen auch unterschiedliche Herangehensweisen und spezielle Auswertungsmethoden, die dann in schwer vergleichbaren Resultaten münden können. Bei einem Vergleich mit ähnlichen Experimenten muss daher sehr genau auf die verschiedenen Parameter geachtet werden. Außerdem ist es auch notwendig für jeden experimentellen Versuch, der glaubhafte Resultate liefern soll, alle wichtigen Parameter zu dokumentieren. Dies dient nicht nur der Nachvollziehbarkeit, sondern kann auch bei darauf aufbauenden Arbeiten auf Fehler hinweisen und Probleme aufdecken. [IT12]

Für vertrauenswürdige Ergebnisse aus Experimenten ist es außerdem wichtig viele Wiederholungen unter gleichen Bedingungen zu realisieren. Dies wird nicht nur durch die mehrfache Durchführung mit verschiedenen Probanden erreicht, sondern auch durch mehrere Wiederholungen der gleichen Testsituation mit dem einzelnen Probanden. Dabei sollten wenigstens zwei, besser noch drei oder vier Wiederholungen das Minimum sein. So können bessere Resultate erzielt, subjektive Schwankungen ausgeglichen und Testpersonen auf ihre Glaubwürdigkeit untersucht werden.

Des weiteren balancieren mehrfache Wiederholungen auch Lern und Kontexteffekte aus. Lerneffekte bewirken, dass innerhalb eines Experiments Wissen durch vorherige Problem-

stellungen vermittelt wird, das für spätere Bewertungen verwendet wird. Ein anderer Mechanismus, mit diesen Lerneffekten umzugehen, ist beispielsweise eine vorherige Trainingseinheit mit ca. 5 kontextrelevanten Stimuli, deren Bewertung nicht in die Gesamtbewertung eingeht [IT08]. Kontexteffekte sind dafür verantwortlich, dass eine Sequenz abhängig von der vorangegangen bewertet wird. So wird häufig eine qualitätsmäßig schlechte Sequenz noch schlechter bewertet, wenn zuvor eine sehr gute Sequenz zu sehen war und umgekehrt [IT12].

Testmaterial

Das Testmaterial ist für den Testzweck sorgfältig auszuwählen. Dabei ist eine ausgewogene Balance zwischen guter und schlechter Qualität in dem Testset wünschenswert, sodass der Gesamtmittelwert der Bewertungen bei einem *Mean Opinion Score (MOS)* von 3 liegen könnte [IT12]. Der MOS ist der Mittelwert aller subjektiven Bewertungen eines Stimuli. Häufig wird dieser dafür im Intervall von 1 bis 5 definiert, wobei 5 die beste Bewertung repräsentiert. Die Reihenfolge der Sequenzen sollte, wenn nicht anders benötigt, zufallsbasiert sein. Da so die zuvor erläuterten Kontexteffekte zusätzlich vermindert werden können. Die Sequenzreihenfolge kann dafür bspw. vom lateinischen Quadrat abgeleitet werden [IT08].

Testdauer

Für die erforderliche Testeinweisung, mögliche Trainingseinheiten und den eigentlichen Test sollte der Proband nicht länger als 30 Minuten benötigen, um das erforderlich Aufmerksamkeitsniveau des Probanden sicherzustellen. Zum Beginn eines Testdurchlaufs können mehrere Probendurchläufe präsentiert werden, damit die Testpersonen eingeübt sind und die bereits erwähnten Lerneffekte abgemildert werden. Sollte das Testverfahren innerhalb eines Experimentes verändert werden, kann jeweils vor dem Wechsel ein entsprechender Probendurchlauf erfolgen. [IT12]

Probanden

Die Anzahl der Testpersonen soll zwischen mindestens 4 und höchstens 40 liegen. In frühen Phasen der Entwicklung und in explorativen Projekten können 4-8 fachkundige Testpersonen in informellen Tests erste aussagekräftige Ergebnisse liefern. Für gewöhnlich werden aber rund 15 Laien empfohlen, die nicht mit solchen Tests und Effekten der Videokodierung vertraut sind. [IT12] [IT08]

Vor dem Test sollen die Testpersonen nachweislich auf Ihre Sehfähigkeiten untersucht werden. Bei einer normalen Sehschärfe sollten keine Fehler in der 20/30 Zeile der Sehtafel nach Snellen (alternativ Landolt) vorkommen. Dafür könnten die Sehtafeln im Idealfall auch in die später genutzte Testumgebung eingepasst werden. Für eine Untersuchung nach möglichen Farbsehstörungen sollten mindestens 10 von 12 Farbsehtafeln nach Beck (alternativ Ishihara) korrekt erkannt werden können. [IT12]

Der Laborbericht sollte außerdem die Fachkenntnis der Testpersonen dokumentieren. In vergleichenden Untersuchungen von verschiedenen Experimenten dieser Art hat man systematische Unterschiede in den Ergebnissen feststellen können. Eine Vermutung ist dabei laut [IT12] das stark unterschiedliche Vorwissen der Testpersonen. Zur Unterstützung dieser Untersuchungen sollen daher Beruf, Geschlecht und Altersgruppe aller Probanden erfasst werden.

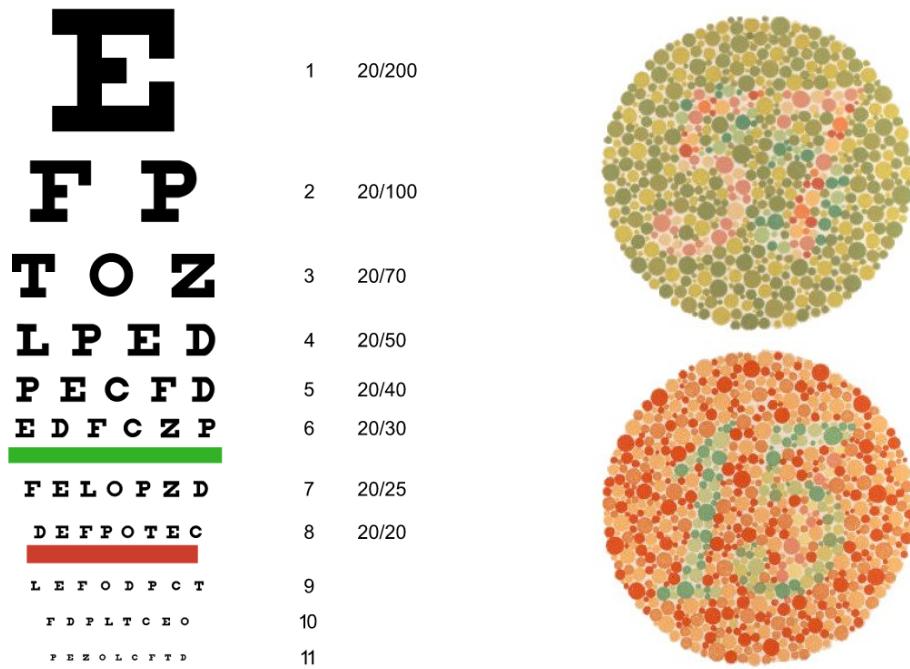


Abbildung 3: Sehtafel nach Snellen [G1] und *Pseudoisochromatic Plates* nach Beck [G2]

Da die Testpersonen elementarer Teil der Messung sind, müssen auch diese vorbereitet werden, um gute Resultate zu bekommen. Vor der Durchführung des Tests sollten die Testpersonen daher eine Beschreibung der Aufgabe bekommen, die mindestens folgende Sachverhalte erklärt:

- Testform: Wie wird getestet?
- Inhalte der Sequenzen und Präsentationsform?
- Was wird bewertet? (Art der auftretenden Fehler oder Qualitätsveränderungen)
- Wie wird bewertet?
- Optional sind Hinweise auf Testdauer, eingeplante Pausen und Probendurchläufe

Die Bandbreite und Sorte der Bildbeeinträchtigungen kann zusätzlich in testunabhängigen Probeversuchen präsentiert werden. Hierfür ist es nicht nötig die Extremfälle zu zeigen und es sollten keine Sequenzen aus dem eigentlichen Test verwendet werden. Es ist darüber hinaus sinnvoll auch Raum für offene Fragen einzuräumen, um frühzeitig auf Fehlinterpretationen reagieren zu können. Außerdem können auch elementare Begriffe wie Bildqualität oder Beeinträchtigung noch einmal eindeutig definiert werden. [IT08]

Technische Realisierung

Für die Gewährleistung einer nachvollziehbaren Testdurchführung ist es außerdem notwendig die technischen Parameter des Experiments zu dokumentieren und diese auch über alle Wiederholungen konstant zu halten. Eine Zusammenfassung der dafür von der ITU in [IT08] empfohlenen Parameter, stellt sich wie folgt dar:

- Maximale Helligkeit des Bildschirms $\approx 200 \text{ cd/m}^2$
- Helligkeitsverhältnis zwischen Vollbild schwarz zu weiß in dunklem Raum ≤ 0.1
- Helligkeitsverhältnis zw. maximaler Bildhelligkeit und Raumhintergrund ≤ 0.2

- Umgebungshelligkeit sollte gering gehalten werden (≤ 20 lux)
- Farbart des Hintergrunds D65 (≈ 6504 Kelvin)
- Ungenutzte Bildflächen sollten mit 50% grau gefüllt werden ($Y=U=V=128$)
- Der maximaler Betrachtungswinkel sollte kleiner als 30° sein.
- Der Betrachtungsabstand vom Bildschirm sollte sich an dem Ziel der Anwendung orientieren [IT08] oder sich nach der PVD-Tabelle (*Preferred Viewing Distance*) richten [IT12].

Screen diagonal (in)		Screen height (H)	PVD	
4/3 ratio	16/9 ratio	(m)	(H)	(m)
12	15	0.18	9	1.62
15	18	0.23	8	1.84
20	24	0.30	7	2.1
29	36	0.45	6	2.7
60	73	0.91	5	4.55
> 100	> 120	> 1.53	3-4	4.59 – 6.12

Abbildung 4: PVD-Tabelle: Demnach berechnet sich der Betrachtungsabstand durch die H-fache Bildschirmhöhe. [IT12]

Der verwendete Monitortyp (CRT, LCD, Plasma, Projektion, etc.) sollte dem Anwendungsfall entsprechend gewählt werden und mit PLUGE (*Picture Line-Up Generation Equipment*) nach [IT07b] kalibriert werden. Dabei sind die spezifischen Eigenschaften der verschiedenen Technologien zu berücksichtigen. [IT08] [IT12]

Dokumentation

Für ein nachvollziehbares Experiment ist es nötig alle wichtigen Parameter der Testumgebung zu erfassen und ausreichend zu dokumentieren. Dafür sind laut [IT12] mindestens die folgenden Informationen notwendig:

- Informationen zur Testkonfiguration
- Informationen über das Testmaterial
- Art der Bildquelle und Informationen über den verwendeten Monitor wie:
Bildschirmgröße, Modellnummer und Konfiguration
- Anzahl und allgemeine Informationen über die Probanden wie:
Altersgruppe, Geschlecht, Berufsgruppe
- Verwendetes Referenzsystem
- Mittelwert der Gesamtbewertung aller Ergebnisse
- Original und modifizierter Mittelwert + Konfidenzintervall 95 %
(Falls eine Testperson aus dem Verfahren ausgeschlossen wird.)

Außerdem sollten nachweisbar Glaubwürdigkeit und Sehfähigkeit der Probanden ermittelt und dokumentiert werden. Eine entsprechend vorgeschlagene Screening-Methode aus [IT12] wird dafür im Abschnitt über die Auswertungsmethoden vorgestellt. [IT12]

2.3.3 Testmethoden

Single Stimulus Methode

Das *Single Stimulus* Verfahren bezeichnet ein Testverfahren, bei dem nur eine einzelne Bildsequenz für den Betrachter zu sehen ist und diese anschließend von ihm in Hinsicht auf eine Fragestellung bewertet wird. Der zeitliche Ablauf stellt sich dafür im Allgemeinen wie folgt dar:

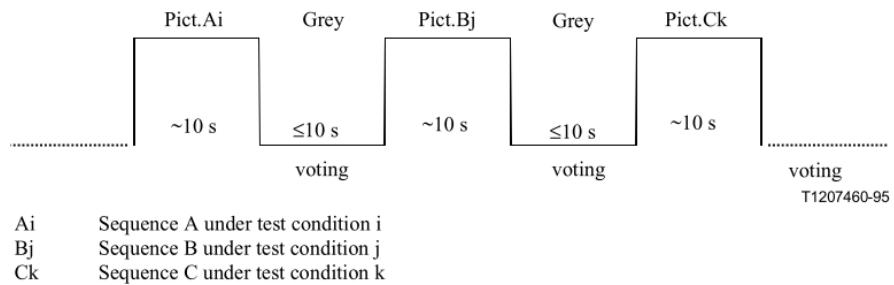


Abbildung 5: Zeitlicher Ablauf der Single Stimulus Methode nach [IT08]

Für diese Methode wird eine Betrachtungszeit von maximal 10 Sekunden und eine Bewertungszeit von weiteren maximal 10 Sekunden vorgesehen. Für *Image Quality Assessments* kann diese Zeit oft erheblich verkürzt werden. Bei der Verwendung von Videosequenzen sollte die Sequenzlänge bei max. 10 Sekunden liegen.

Die Bewertung erfolgt im Anschluss an jeder Sequenz durch ein sinnvolles Bewertungsschema. Im Zusammenhang mit der Verwendung der fünfstufigen ITU-Skala wird dieses Verfahren häufig als *Absolut Category Rating (ACR)* oder *Adjectival Categorical Judgment* Methode bezeichnet [IT08]. Eine genaue Beschreibung dieser und anderer Bewertungsmethoden erfolgt im Anschluss an die Testmethoden.

Hidden Reference Ein erweitertes Konzept stellt hier die *Hidden Reference* Methode dar. Dabei beinhalten der Test zusätzlich auch die unbeeinträchtigten Referenzbilder, ohne dass die Probanden davon wissen (*hidden*). Dadurch ist es möglich in der anschließenden Auswertung den DMOS (*Differential Mean Opinion Score*) zwischen Referenz- und Testbild zu berechnen: $DMOS(seq) = MOS(test_{seq}) - MOS(orig_{seq}) + 5$. Ein DMOS von 5 repräsentiert die beste Bildqualität, während der Wert von 1 eine mangelhafte Qualität anzeigt. Ein Wert > 5 bedeutet, dass die Referenz schlechter bewertet wurde als das Testbild. In diesem Fall muss für eine spätere Auswertung der Wert transformiert werden, um die Gesamtbewertung nicht zu verfälschen. Ein großer Vorteil dieses Konzepts ist die zusätzliche Möglichkeit zur Glaubwürdigkeitsprüfung der Probanden. [IT08]

Single Stimulus with Multiple Repetition: In [IT12] wird außerdem das Konzept des *Single Stimulus with Multiple Repetition* (SSMR) beschrieben. Dabei sollte innerhalb des Versuches dasselbe Testset 3-mal von einem Probanden durchlaufen werden. Dafür gilt, dass die Bildreihenfolge in den drei Durchgängen unterschiedlich ist und außerdem auch keine Bildpaarfolge wiederholt wird. Der erste Durchgang wird dabei lediglich für die Eingewöhnung genutzt und seine Resultate werden verworfen. Nur die Ergebnisse aus dem zweiten und dritten Durchgang werden anschließend berücksichtigt. Das Ziel dieser Methode ist eine Reduzierung des Kontexteffekts. [IT12]

Double Stimulus Methode

Bei der *Double Stimulus* Methode wird dem Probanden sowohl das Testbild und auch das dazugehörige Referenzbild präsentiert. Dies kann sequenziell oder parallel umgesetzt werden. Die anschließende Bewertung wird dabei, laut [IT08], ausschließlich für das Testbild abgegeben, wobei im Falle einer sequenziellen Präsentation das Referenzbild lediglich im Gedächtnis des Probanden vorhanden ist. Der zeitliche Ablauf für eine Variante des *Double Stimulus* Verfahrens wird wie folgt empfohlen:

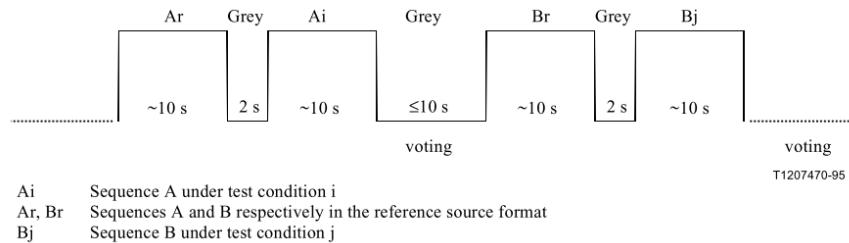


Abbildung 6: Zeitlicher Ablauf für das *Double Stimulus* Verfahren nach [IT08]

Wie in Abb. 6 dargestellt wird nach einer Stimulationszeit des Originals von max. 10 Sekunden eine kurze Pause von 2 Sekunden eingeplant. Nach der anschließenden Testsequenz mit einer maximalen Dauer von weiteren 10 Sekunden folgt die Bewertung der Testsequenz. Auch hier können für *Image Quality Assessments* kürzere Zeiten eingeplant werden. Bei einer anschließenden Bewertung mit Hilfe der ITU-Skala wird dieses Verfahren häufig in der Literatur als *Double Stimulus Impairment Scale* (DSIS) oder als *Degradation Category Rating* (DCR) Verfahren bezeichnet. [IT08]

Der typische Anwendungsfall für DSIS ist die Bewertung eines neuen Systems oder die Beurteilung eines Übertragungsfehlers. Die Reihenfolge der verschiedenen Sequenzpaare kann dabei willkürlich gewählt werden. Besonders bei Sequenzen mit sehr feinen Unterschieden ist hier eine Wiederholung der Sequenzen empfohlen, da somit wesentlich bessere Resultate erzielt werden können [IT12]. Ein beispielhafter Aufbau eines zeitgesteuerter DSIS-Assessments kann der Skizze in Abb. 7 entnommen werden.

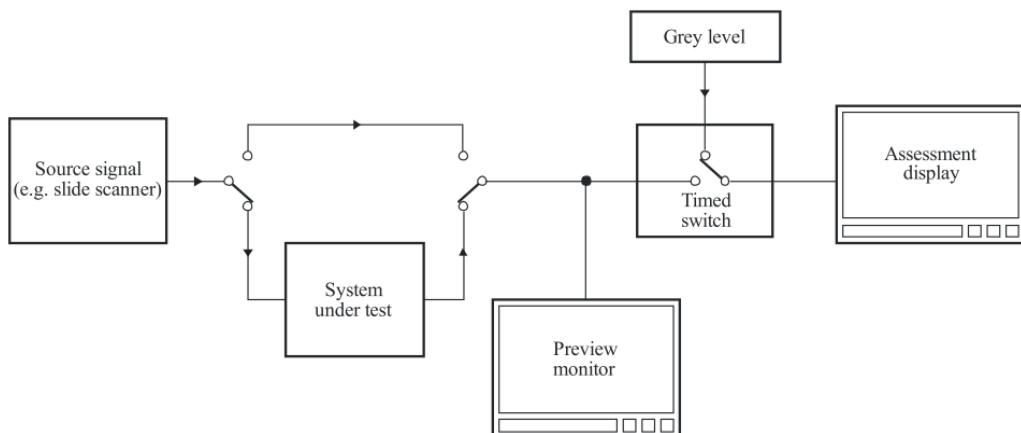


Abbildung 7: Eine zeitbasierte Mechanismus steuert die Präsentation von Referenz, Stimuli und der Graustufen als Pauseneinblendung. [IT12]

Weitere Varianten des *Double Stimulus* Assessment können in [IT12] gefunden werden. Bei diesen Verfahren werden dem Betrachter wie zuvor Test- und Referenzsequenz präsentiert. Abhängig von der Personenanzahl werden hier allerdings zwei unterschiedliche Testmöglichkeiten beschrieben.

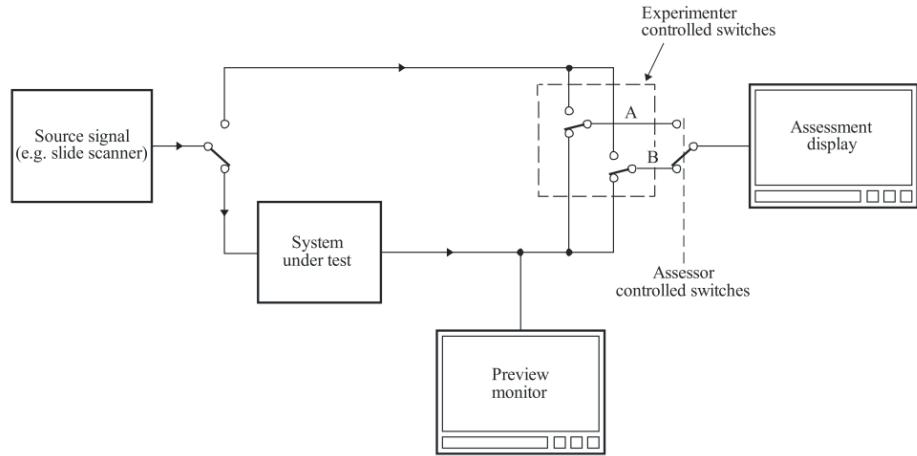


Abbildung 8: Aufbauskizze für die Steuerungsvarianten I und II [IT12]

Variante I

Eine Variante ist, dass der Proband zwischen Original und Testbild beliebig wechseln darf, bis sich seine Meinung zu den Sequenzen gefestigt hat. Anschließend werden beide Bilder bewertet. Es sollte dem Probanden aber währenddessen nicht bekannt sein welches Bild das Original und welches das Testbild ist. (Bildschirmanordnung der Sequenzen muss zufallsbasiert wechseln) [IT12]

Variante II

Die andere Variante ist für mehrere gleichzeitige Betrachter konzipiert. Dabei wird ihnen mehrfach für dieselbe Zeitspanne das gleiche Sequenzpaar präsentiert. Anschließend erfolgt in einem letzten Durchlauf die Bewertung. Bei Einzelbildern werden dafür fünf Wiederholungen für je 3-4s empfohlen, wobei während der letzten zwei Wiederholungen bewertet werden soll. Bei Videosequenzen werden zwei Durchläufe empfohlen, bei denen die Bewertung während der zweiten Durchführung erfolgt. Auch hier sollten Videosequenzen nicht länger als 10 Sekunden dauern. [IT12]

Für die Bewertung werden zwei kontinuierlicher Schieberegler empfohlen, da diese eine Vermeidung von Quantisierungsfehlern ermöglichen. Dabei sollte die Skala in die 5 Bewertungskategorien der ITU-Skala eingeteilt werden. Die Anwendung der *Double Stimulus* Methode mit einer kontinuierlichen ITU-Qualitätsskala wird häufig auch als *Double Stimulus Continuous Quality Scale (DSCQS)* Methode bezeichnet.

Stimulus Comparison / Pair Comparison Methode

Bei den *Stimulus Comparison* (oder *Pair Comparison*) Methoden werden die Sequenzen paarweise präsentiert. Dabei können entweder Bildpaare aus Referenz- und Testbild verwendet oder auch zwei Testbilder miteinander verglichen werden. Eine Anwendung mit zwei beeinträchtigten Testbildern ist bspw. der Qualitätsvergleich verschiedener Kodierverfahren. Die Bewertung findet jeweils nach einem Sequenzpaar statt und sollte lediglich die Entscheidung beinhalten, welche Sequenz eine bessere Qualität aufweist. Bei dem *Pair Comparison* Verfahren sollten immer alle Kombinationsmöglichkeiten $n \cdot (n - 1)$ der Sequenzen präsentiert werden, was sehr zeitaufwendig ist. In [IT08] und [IT12] werden dafür drei Arten von *Stimulus Comparison* Verfahren vorgestellt:

Adjectival Categorical Judgement Method

Der Betrachter bewertet das Verhältnis der beiden Bilder eines Sequenzpaars im Bezug auf eine Fragestellung. Dafür kann bspw. der Unterschied zwischen den Bildern erfragt werden (gleich, ungleich) oder auch das Vorhandensein und die Richtung der Beeinträchtigungen (schlechter, gleich, besser)

Non-categorical Judgement Method

Der Betrachter vergibt einen Wert für die Beziehung zwischen dem Bildpaar wie bisher, nur auf einer anderen Skala:

Kontinuierliche Skala (Same \leftrightarrow Different) oder

Note im Bezug auf eine Eigenschaft (Bspw. von 0 bis 10)

Performance Judgement Method

Der Betrachter entscheidet lediglich anhand einer Fragestellung, für welches Bild er sich entscheidet. Mögliche Fragestellungen könnten sein: Welches Bild ist qualitativ besser? In welchem Bild sind mehr Störungen zu erkennen? In welchem Bild ist Störung xy stärker zu sehen? Welches Bild gefällt Ihnen besser?

Synchrone Betrachtung

Sollte es für das Testsystem möglich sein, so wird für *Double Stimulus* und *Stimulus Comparison* Methoden empfohlen eine gleichzeitige Präsentation der Sequenzen zu realisieren. Falls dafür zwei Monitore benötigt werden, müssen diese sich in allen Parametern gleichen. Es sollten zusätzlich Testfälle entwickelt werden, um die Gleichheit der Monitore zu prüfen. Die Vorteile dieser Darstellungsmethode werden in [IT08] wie folgt aufgeführt:

- Enorme Reduzierung der Testdauer
- Höhere Aufmerksamkeit durch die kürzere Testdauer
- Qualitätsunterschiede können besser wahrgenommen werden

Unter Berücksichtigung besonderer Vorkehrungen wie:

- Perfekte Synchronisierung der Sequenzen
- Einen Abstand von der 8-fachen Bildhöhe um die Augenbewegung zwischen den Sequenzen zu reduzieren (Bspw. 2.40 m Abstand bei Bildhöhe von 30 cm)
- DCR: Referenzbild mit fester Seite und Testperson darauf hinweisen
- PC: Alle kombinatorischen Möglichkeiten der Seitenaufteilung präsentieren

Subjective Assessment of Multimedia VIDEo Quality

Beim SAMVIQ-Verfahren handelt es sich um ein weiteres Testverfahren der ITU, dass sich dadurch auszeichnet dem Probanden mehr Kontrolle über den Testablauf zu geben. Dafür bekommt dieser eine Steuerungsmöglichkeit durch die er Zugriff auf verschiedene Versionen einer Sequenz hat. Er kann dabei beliebig oft zwischen den Sequenzen wechseln und seine Bewertung dafür abgeben oder korrigieren. Dabei hat der Proband auch Zugriff auf eine explizite Referenz, die außerdem auch als *Hidden Reference* zusätzlich unter die Testsequenzen gemischt werden kann. Bei der Präsentation sollte beachtet werden, dass die verschiedenartigen Beeinträchtigungen nicht immer mit dem selben Steuerelement verknüpft sind, sondern wie in Abb. 10 unterschiedlich angeordnet. Für die Bewertung wird außerdem eine kontinuierliche ITU-Skala empfohlen. [IT07a]

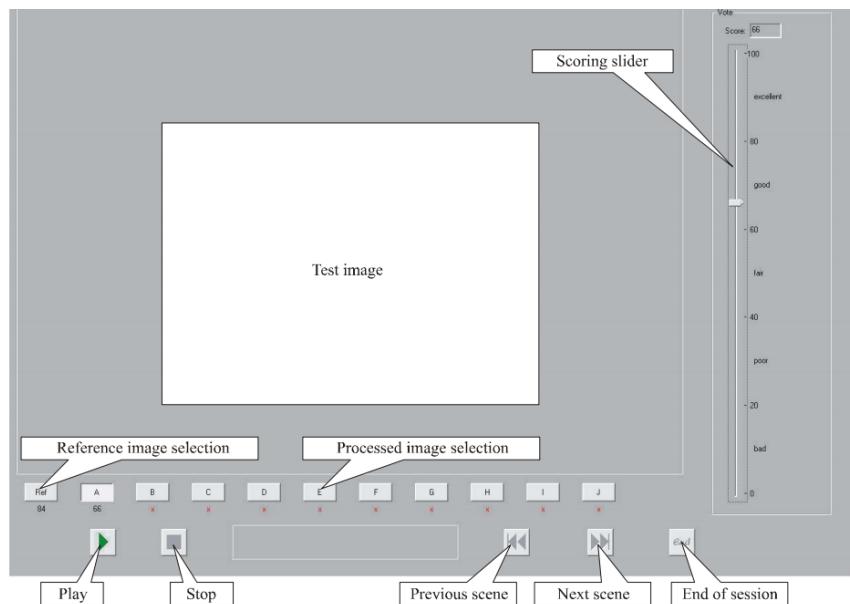


Abbildung 9: Vorschlag für die Umsetzung des SAMVIQ-Verfahrens mit diversen Steuerelementen [IT07a]

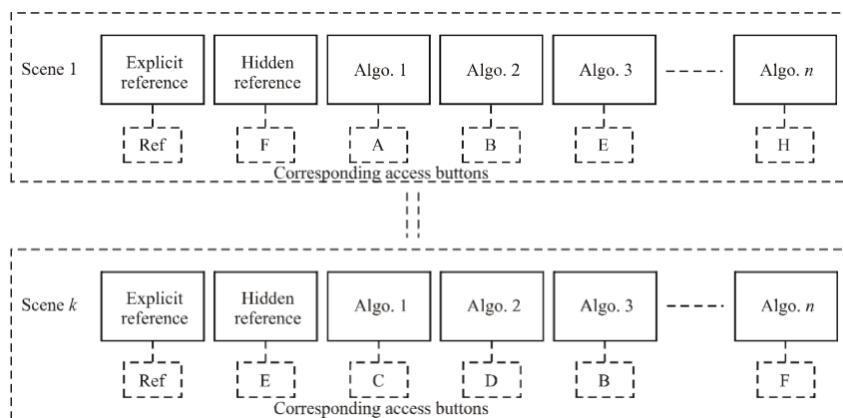


Abbildung 10: Vermeiden einer Verknüpfung zw. Steuerelement und Beeinträchtigung. [IT07a]

2.3.4 Bewertungsmethoden

Categorical Judgement

Für die Bewertung der Bildqualität werden zu jeder Methode entsprechende Bewertungsskalen empfohlen. Eine übliche Skala ist dabei die ITU Qualitäts- und Beeinträchtigungsskala. Wie sich herausgestellt hat, ist diese Bewertung stabiler bei kleinen Unterschieden als bei großen. [IT12]

Five-grade scale	
Quality	Impairment
5 Excellent	5 Imperceptible
4 Good	4 Perceptible, but not annoying
3 Fair	3 Slightly annoying
2 Poor	2 Annoying
1 Bad	1 Very annoying

Abbildung 11: ITU Qualitäts- und Beeinträchtigungsskala [IT12]

Diese kann aber unter bestimmten Voraussetzungen ungeeignet sein oder nicht ausreichen. Sind kleinere Unterschiede zwischen den Schritten sinnvoll, kann die Skala um 4 oder 6 weitere Schritte (Abb. 12) ergänzt werden. Die entsprechend 9 oder 11-Stufige Skala hat sich besonders bei Assessments mit Kodierverfahren für niedrigen Bitraten als sehr nützlich erwiesen [IT08]. Sollte es möglich sein, kann die Skala auch kontinuierlich realisiert werden. Die Ergebnisse sollten dafür auf ganzzahlige Werte im Intervall 0 - 100 diskretisiert werden. [IT12]

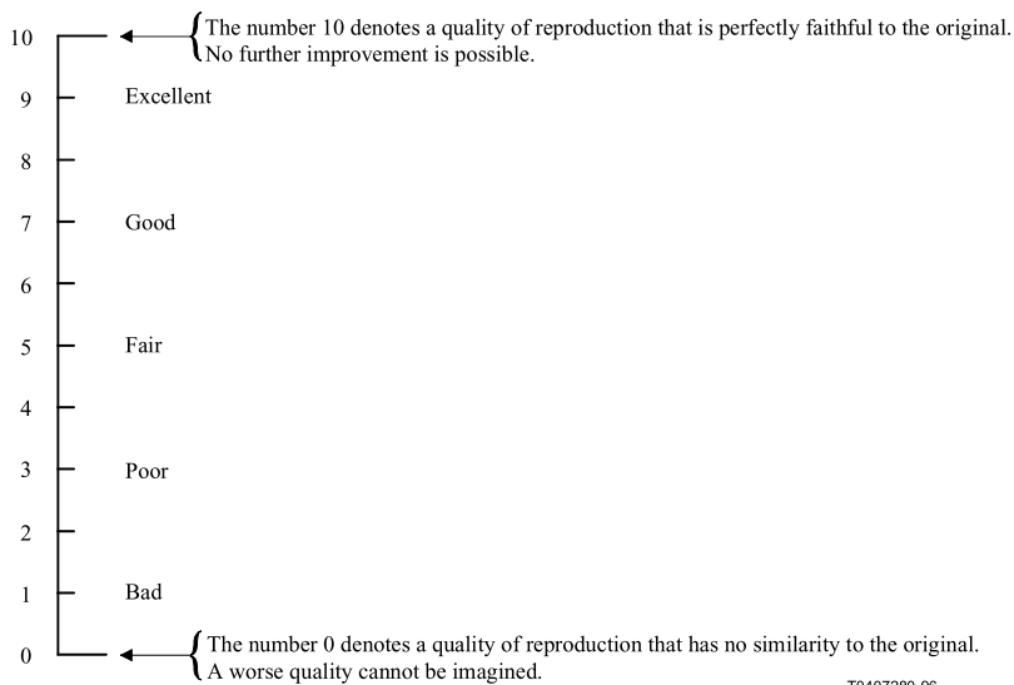


Abbildung 12: ITU Bewertungsskala mit 11 oder 9 Stufen (ohne 0 und 10) [IT08]

Non-categorical Judgement

Ein wichtiger Punkt der Bewertungsmethode ist die Skalenbezeichnung, da diese unter Umständen zu sprachlichen Missverständnissen führen kann. Besteht diese Gefahr, kann eine einfachere Skala verwendet werden, die lediglich zwei Bezeichnungen an den Enden hat und über eine Mittelpunktmarkierung verfügen sollte.

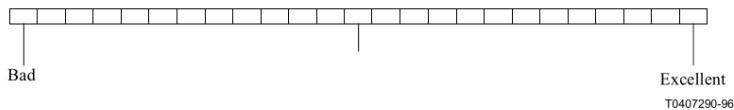


Abbildung 13: Non-categorical judgement Skala nach [IT08]

Numerical Categorical Judgement

Eine grundsätzlich andere Bewertungsmethode ist die Bewertung mit Zahlen. Dabei sollte für die entsprechende Fragestellung ein entsprechendes Intervall vorgegeben werden. Die *Single Stimulus* Methode im Zusammenspiel mit numerischer Bewertung stellt sogar, laut ITU, die robusteste und sensitivste *No-reference* Bewertungsmethode dar [IT12]. Falls Testpersonen Zahlenwerte angeben sollen, kann es hilfreich sein sie zu ermutigen Nachkommastellen zu verwenden, um so präzisere Resultate zu erhalten. Bewährte Intervalle für dieses Bewertungstechnik sind: 1 - 5, 0 - 10 oder 0 - 100. [IT12]

Weitere Bewertungsverfahren

Es existiert außerdem noch eine Menge an weiteren Bewertungsverfahren, wie das bereits erwähnte *Adjectival Categorical Judgement* oder die *Performance Judgement Method*. Aufgrund der Vielzahl soll aber an dieser Stelle auf Vollständigkeit verzichtet werden und auf einen anderen Punkt der Bewertungsverfahren eingegangen werden. Sollte es der Fall sein, dass die Resultate in einem Assessment sich sehr ähneln, wird eine weitere Ausdifferenzierung der Fragestellung zu einzelnen Bildeigenschaft empfohlen. Dies kann zusätzliche Informationen über Probleme des Experiments oder des Kodierverfahrens liefern, die durch simples Bewerten der Gesamtqualität nicht erfasst werden. So wird bspw. eine durchgehend schlechte Helligkeit zu einer permanenten Verschlechterung der Ergebnisse führen, die unentdeckt bleiben könnte. Erst eine detaillierte Befragung nach der Helligkeit würde auf eine permanente Störung aufmerksam machen. Weitere Bildeigenschaften, die dafür bewertet werden könnten sind bspw. Kontrast, Farbechtheit, Konturen, *Jerkiness* und viele Weitere [IT08]. Im Gesamten können auch so, durch die gewichtete Addition von Einzeleffekten, gute Aussagen über die Qualität der Sequenzen erarbeitet werden und zusätzlich systematische Fehlerquellen aufgedeckt werden. [IT08]

Vergleichbarkeit

Das gewählte Bewertungsverfahren hat dabei eine enorme Auswirkung auf die Vergleichbarkeit einzelner Experimente untereinander. So sollte bspw. keine Umrechnen von numerischer in kategorische Bewertung erfolgen, da hier unterschiedliche Interpretationen der Probanden hineinspielen [IT12]. Auch abweichende Skalenbezeichnungen in Experimenten führen unmittelbar dazu, dass eine Vergleichbarkeit nicht mehr möglich ist, wie es auch bei unterschiedlich ausgeprägten Kontexteffekten und abweichenden Akzeptanzkriterien der Fall sein kann. [IT08]

2.3.5 Auswertungsmethoden

Zur sinnvollen und glaubwürdigen Auswertung der erhobenen Daten ist die Angabe von Mittelwert, Standardabweichung und Konfidenzintervall (95%) üblich. Dieses Vorgehen sollte auch in der grafischen Darstellung berücksichtigt werden, indem zusätzlich zu den Durchschnittswerten auch der Toleranzbereich durch die Angabe des Konfidenzintervalls wie folgt durchgeführt wird: [IT12]

Mitteln aller Individualbewertungen:

$$\bar{u}_{jkr} = \frac{1}{N} \sum_{i=1}^N u_{ijkr}$$

Wobei u_{ijkr} die Bewertung durch Probanden i von Sequenz k mit der Beeinträchtigung j und der Wiederholung r darstellt und N die Anzahl der Probanden repräsentiert.

Angaben des Konfidenzintervalls CI:

$$CI = [\bar{u}_{jkr} - \sigma_{jkr}, \bar{u}_{jkr} + \sigma_{jkr}]$$

$$\sigma_{jkr} = 1.96 \frac{S_{jkr}}{\sqrt{N}}$$

Mit S_{jkr} als Standardabweichung:

$$S_{jkr} = \sqrt{\sum_{i=1}^N \frac{(\bar{u}_{jkr} - u_{ijkr})^2}{N-1}}$$

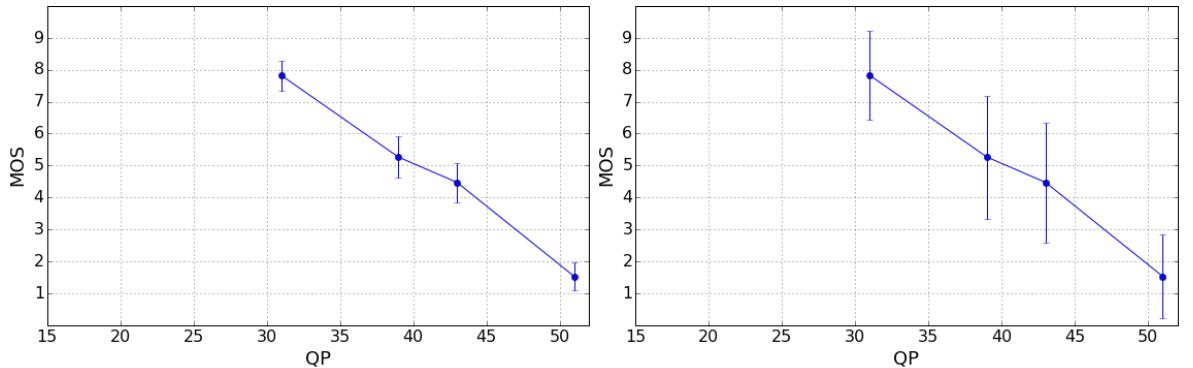


Abbildung 14: Graphen einer Sequenz mit vier verschiedenen Qualitätsstufen, die von Probanden bewertet wurden. Mit Angabe vom Konfidenzintervall (95%) links und der Standardabweichung rechts. [G3]

Weitere informative Grafiken stellen außerdem, laut [IT08], Histogramme der Bewertungen dar. Anhand dieser können häufige Fragestellungen schnell beantwortet und Tendenzen gut erkannt werden. [IT08]

Eine weitere vielgenutzte Visualisierungstechnik in der Literatur sind Streudiagramme. Sie sind sehr gut geeignet, um die Korrelation zwischen Metriken und subjektiven Bewertungen darstellen.

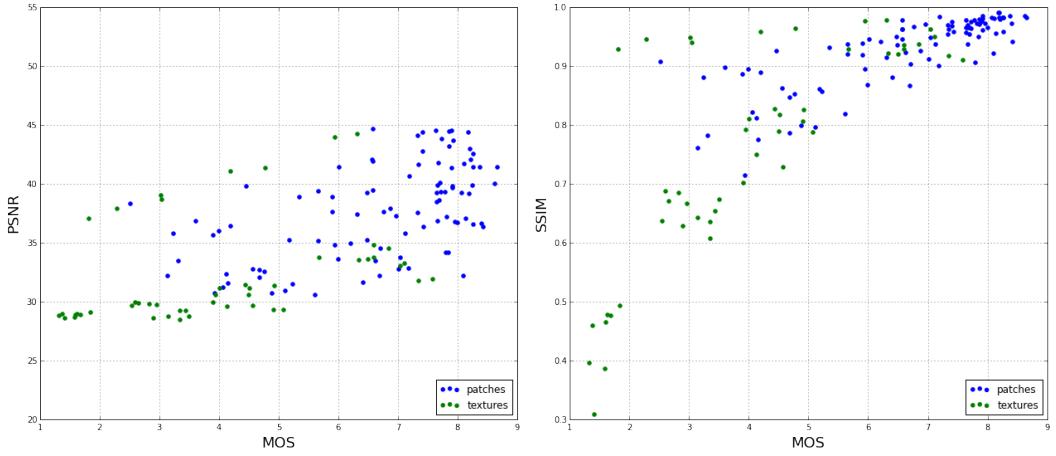


Abbildung 15: Streudiagramme für eine Korrelationsbetrachtung zwischen PSNR und MOS (links) sowie SSIM und MOS (rechts). [G4]

Ein letzter wichtiger Punkt für die über die Daten ist, dass die Ergebnisse auch nutzbar für die Weiterentwicklung und Überprüfung gemacht werden. Dafür schlägt die ITU noch einige gemeinsame Konventionen für Datensätze vor. [IT12]

Probandenscreening

Des weiteren sollte immer Untersuchungen über die Glaubwürdigkeit der Probanden durchgeführt werden. Dafür wird, nach ITU, zunächst die Bewertungsverteilung darauf untersucht, ob sie normalverteilt ist. Das vorgeschlagene Verfahren verwendet dafür den β_2 -Tests. Ergibt β_2 einen Wert im Intervall $[2 \leq \beta_2 \leq 4]$ wird die Verteilung als normalverteilt angenommen. Anschließend werden die Individualbewertungen mit den Gesamtbewertungen eines Stimuli verglichen: [IT12]

$$\beta_{2jkr} = \frac{m_4}{(m_2)^2}; m_x = \frac{\sum_{i=1}^N (u_{ijkr} - \bar{u}_{ijkr})^x}{N}$$

Gesucht wird P_i und K_i für den User i über alle $j, k, r = 1$ bis J, K, R
(Alle Sequenzen K mit Beeinträchtigungen J und Wiederholungen R)

Wenn $2 \leq \beta_{2jkr} \leq 4$, dann:

$$\begin{aligned} \text{Wenn } u_{ijkr} \geq \bar{u}_{ijkr} + 2S_{jkr}, \text{ dann } P_i &+= 1 \\ \text{Wenn } u_{ijkr} \leq \bar{u}_{ijkr} - 2S_{jkr}, \text{ dann } Q_i &+= 1 \end{aligned}$$

sonst:

$$\begin{aligned} \text{Wenn } u_{ijkr} \geq \bar{u}_{ijkr} + \sqrt{20}S_{jkr}, \text{ dann } P_i &+= 1 \\ \text{Wenn } u_{ijkr} \leq \bar{u}_{ijkr} - \sqrt{20}S_{jkr}, \text{ dann } Q_i &+= 1 \end{aligned}$$

Falls $\frac{P_i+Q_i}{JKR} > 0.05$ und $|\frac{P_i-Q_i}{JKR}| < 0.03$ dann sollte der Proband i entfernt werden.

2.4 Zusammenfassung

Die vorgestellten Möglichkeiten zur Bewertung von Bild- und Videosequenzen lassen bisher eine wichtige Frage offen: Welche Berührungs punkte haben Image Quality Assessments und Image Quality Metriken? Eine Vielzahl an Untersuchungen, wie [DY09], [WB09], [LJ11] und [VQE10], haben dahingehen bereits gezeigt, dass die Resultate aus Assessments und Metriken erheblich voneinander abweichen können. Sie enthalten darin, wie anfällig selbst etablierte Verfahren sind und kritisieren das leichtsinnige Vertrauen in diese Metriken. Selbst PVQMs, die unter Berücksichtigung von wahrnehmbaren Charakteristiken entwickelt wurden, schneiden bei diesen Vergleichen mit subjektiven Bewertungen häufig nur wenig besserer ab [DY09].

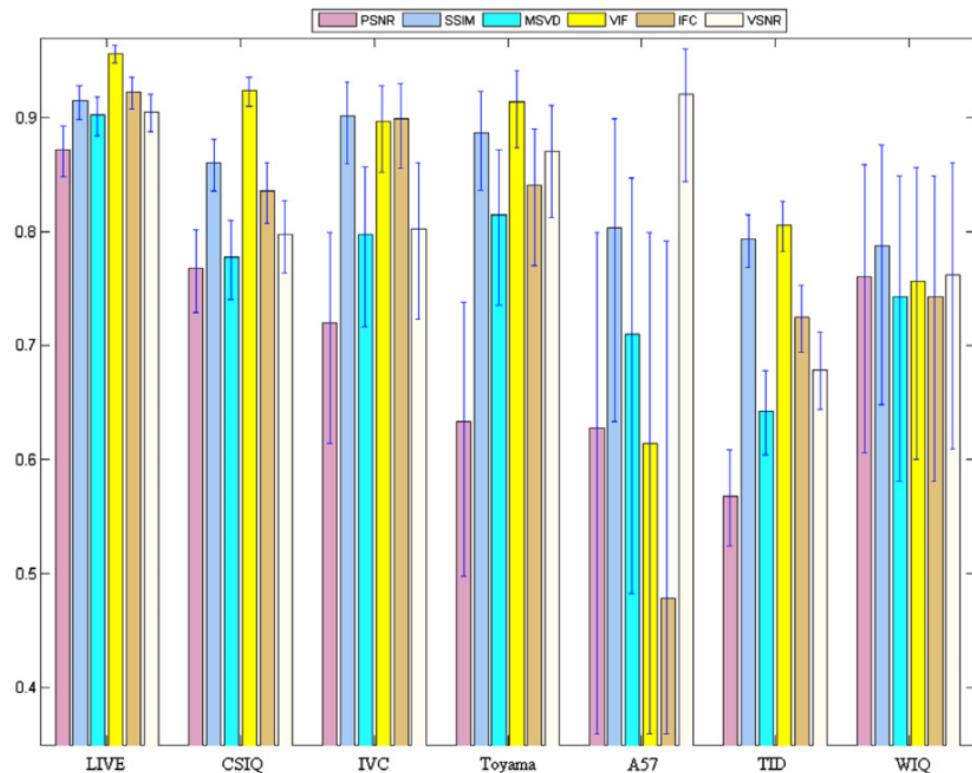


Fig. 5. C_p of different metrics for various databases (with 95% CI indicated).

Abbildung 16: Pearson-Korrelation mit CI (95%) von 6 Metriken mit Bewertungen aus diversen Datenbanken mit subjektiven Bewertungen. [LJ11]

Wie in Abb. 16 zu sehen entsteht unter den richtigen Bedingungen der Anschein, dass einzelne Metriken dem menschlichen Sehvermögen gut entsprechen. Bei abweichenden Testbedingungen durch die Verwendung anderer Datenbanken, wie A57, TID und WIQ, ergibt sich allerdings eine viel schlechtere Korrelation. [LJ11]

Mit dieser Erkenntnis sollte die Beantwortung aktueller Forschungsfragen nicht nur auf Grundlage von Metriken erfolgen. *Image Quality Assessments* sind hier ein unerlässliches Mittel zur Plausibilitätsprüfung und müssen weiterhin für die Validierung von neuen und bestehenden Metriken eingesetzt werden. Diese Tatsache ist, wie Eingangs bereits erläutert, ein wesentlicher Grund für die weiterhin notwendige Beschäftigung mit Image Quality Assessments und darüber hinaus eine Motivation für die geplante Implementierung.

3 Konzept

Ziel dieses Kapitels ist es, die Herangehensweise an das Projekt darzulegen und elementare Entscheidungen und Lösungskonzepte nachvollziehbar zu machen.

3.1 Crowdsourcing

Ein vielversprechender und zeitgemäßer Ansatz im Bereich der IQAs sind webbasierte Applikationen, die eine Teilnahme am Assessment über das Internet ermöglichen. Das sogenannte *Crowdsourcing* beschreibt dabei den Auslagerungsprozess (*outsourcing*) einer bisher selbst erbrachten Leistung auf eine große Anzahl von Menschen über das Internet (*crowd*). Verschiedene Projekte, wie *CrowdMOS*, *QualityCrowd2* und *in-momento*, haben hier schon unterschiedliche Lösungen für die Entwicklung solcher Anwendungen präsentiert [TH14]. Leider sind viele dieser Anwendungen unvollständig und häufig nur für einzelne Testverfahren entwickelt worden. Da eine Erweiterung dieser Systeme mehr Einarbeitungszeit benötigen würde als eine eigenständige Entwicklung, soll ein alternatives Projekt dazu entstehen. Eine Diskussion der Vorteile und möglicher Probleme von webbasierten Anwendungen soll hier als Basis für die Entwicklung des Konzeptes dienen:

Vorteile	Probleme
Viele und vielfältige Teilnehmer	Unkontrollierte und vielfältige Fehlerquellen
Parallele Durchführung mehrerer Assessments	Schlechte Betreuung der Teilnehmer
Geringerer organisatorischer Aufwand	Geringere Teilnehmermotivation
Plattformunabhängigkeit und zentrale Wartbarkeit	Technischer Aufwand, Realisierbarkeit und Sicherheit

Die klaren Vorteile des *Crowdsourcing* sind die enormen Zeit- und Kostenersparnisse. Idealerweise können Probanden dabei gleichzeitig ihre webbasierten Assessments absolvieren und in ihrer gewohnten Umgebung bleiben. Außerdem können damit Teilnehmer auf der gesamten Welt adressiert werden. Dadurch ist nicht nur eine enorme Erweiterung der potentiellen Teilnehmern möglich, sondern auch eine große Meinungsvielfalt durch Menschen mit unterschiedlichen Vorraussetzungen und abweichenden Gewohnheiten. Der damit verbundene organisatorische Aufwand bleibt dabei vergleichsweise klein. Des weiteren sind klare technische Vorteile zu nennen, wie eine zentrale Wartbarkeit und eine naturgemäß Plattformunabhängigkeit der Webtechnologien, die außerdem gut dokumentiert und weit verbreitet sind. Um den technischen Aufwand dabei gering zu halten, kann auf eine Vielzahl an bewährte Lösungen zurückgegriffen werden, die eine grundlegende Infrastruktur weitestgehend bereitstellen.

Ein klarer Nachteil einer *crowdbasierten* Anwendung ist dabei die unkontrollierte Umgebung. Dabei verringern vielfältige Fehlerquellen und Missverständnisse deutlich die Akzeptanz eines solchen Assessments. Dafür ist unter anderem auch die fehlende Probandenbetreuung verantwortlich, die zu noch mehr Missverständnissen und Fehlannahmen führen wird. Zudem wird vermutlich auch eine geringere Teilnehmermotivation zu erkennen sein, die vermehrt zu frühzeitigen Abbrüchen führen wird. Dieser Sachverhalt muss dabei durch eine größere Probandenzahl ausgeglichen und bei der anschließenden Datenanalyse umbedingt beachtet werden. Bei umfangreicheren Projekten dieser Art, sind

außerdem Themen wie Skalierbarkeit und Datensicherheit für die Bereitstellung eines solchen Webdienstes von großer Bedeutung. Ein weiterer Nachteil ist, dass auf Grundlage des geringen *Quality of Service* bei Webanwendungen eine ideale Realisierung von Video-Assessments nicht möglich ist. Schwankende Bandbreiten, Delays und mangelnde Echtzeitfähigkeit können zwar reduziert werden, aber werden in einem weltweiten Kontext immer Probleme bereiten, die auch schon bei Einzelbildern zu berücksichtigen sind.

Funktionale Anforderungen:

Unter funktionalen Gesichtspunkten soll die Anwendung die entsprechenden Bewertungs- und Testverfahren aus den ITU-Richtlinien enthalten, dazu gehören:

- Single Stimulus Assessments (SS)
- Double Stimulus Assessments (DS)
- Pair Comparison Assessments (PC)
- Subjective Assessment of Multimedia Video Quality (SAMVIQ)

Außerdem sollen hierbei die vorgeschlagenen Bewertungsverfahren nutzbar sein, wie Schieberegler mit verschiedenen Skalen, numerische Bewertung oder Sequenzauswahl. Eine gewisse Modularität soll dabei die unterschiedlichen Kombinationsmöglichkeiten aus Testverfahren und Bewertungsmechanismus ermöglichen. Die Verfahren werden zunächst für Einzelbilder entwickelt und anschließend für Testzwecke prototypisch in Video-Assessments umgebaut.

Weitere funktionale Anforderungen lassen sich außerdem wie folgt auflisten:

- Verwaltung von Nutzersitzung
- Formularbasierte Nutzerdatenabfrage
- Erstellung von Assessments mit diversen Faktoren wie:
Hidden References, Trainingseinheiten, zufallsbasierte Sequenzanordnung, gleichzeitige Betrachtung und Bewertung der Stimuli, etc.

Die Realisierung der Webanwendung wird dabei mit dem frei verfügbaren Web Application Framework *Symfony 2* vorgenommen. Dieses Framework bringt eine Vielzahl an nützlichen und benötigten Features, wie einen integrierten Webserver, intelligente Datenbankanbindung, Templatemechanismen, Lokalisierungsfunktionen und vieles mehr.

3.2 Datenbank

Das zweite wichtige Konzept ist, dass eine Datenbank alle relevanten Informationen über die Assessments enthalten soll. Damit soll sichergestellt werden, dass alle Faktoren der Experimente reproduzierbar sind und gut nachvollzogen werden können. Aus dieser Datenbank soll somit auch eine vollständige Analyse des Experimentes möglich sein. Die in [IT12] vorgeschlagenen Konventionen für die Datensätze bilden leider nur die von der ITU vorgeschlagenen Verfahren geeignet ab. Als Datenbank dient der Applikation daher eine selbst entworfene Datenbank. Das hierfür vorgesehene Datenbankmanagementsystem ist MySQL, da dies ein Open-Source Projekt ist, dass über ein relationales Datenbankmodell verfügt. Außerdem gestaltet sich die Anbindung der MySQL-Datenbank in das *Symfony* Framework sehr einfach.

Die Konfiguration eines Experiments, die Erstellung eines neuen Testverfahrens oder das Hinzufügen neuer Testsequenzen soll später direkt in der Datenbank realisiert werden. Das ist zwar zunächst nicht besonders komfortabel, aber ausreichend, da die Applikation nur für den internen Gebrauch am HHI vorgesehen ist. Die Entwicklung einer benutzerfreundlichen Oberfläche erzeugt hier mehr Aufwand als Nutzen.

3.3 Datenanalyse

Ein erheblicher Teil bei der Durchführung von Assessments dieser Art besteht in der sinnvollen Auswertung und Visualisierung der erhobenen Messdaten. Da dies eine tägliche Problemstellung in Wissenschaft und Technik ist, soll an dieser Stelle auf eine etablierte Funktionsbibliothek zurückgegriffen werden. Diese Funktionalität wird daher durch die Verwendung diverser Module der Programmiersprache Python, wie *numpy*, *pandas* und *matplotlib* umgesetzt. Je nach Ziel und Umsetzung eines Experiments ergeben sich dabei individuellen Aufgaben für die Auswertung. Eine Grundmenge an Funktionalität kann dennoch aus den Beschreibungen in [IT12] entnommen werden, wie beispielsweise:

- Mittelwert aller Bewertungen
- Mittelwert aller Bewertungen eines Stimuli
 - + Standardabweichung
 - + Konfidenzintervall
- Diverse Histogramme der Bewertungen
- Diverse Korrelationsbetrachtungen wie MOS - Metrik
- Glaubwürdigkeitsuntersuchung der Probanden

3.4 Architektur

Für die Architektur der Anwendung soll dabei das 3-Tier Client-Server Modell dienen. Die Anwendung wird dafür in drei Schichten strukturiert. Die Präsentationsebene ist zuständig für das User Interface und stellt die Kommunikationsschnittstelle zwischen Anwender und Anwendung dar. Die Anwendungsebene enthält die eigentliche Programmlogik, führt Berechnungen aus und ist Vermittler der Kommunikation zwischen Präsentations- und Datenebene. Diese Datenebene ist für die Verwaltung der Daten zuständig und stellt diese für die Anwendungsebene zur Verfügung. Im Client-Server Modell entspricht der Client dabei der Präsentationsebene (*Presentation Tier*), während Anwendungs- und Datenebene (*Logic- und Data Tier*) den Server repräsentieren.

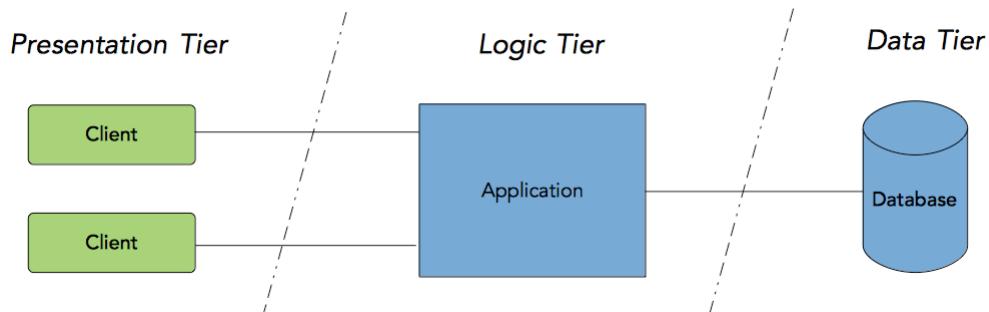


Abbildung 17: 3-Tier Client-Server Modell [G5]

Der Vorteil dieser Architektur ist, dass viele User gleichzeitig den vom Server angebotenen Dienst nutzen können. Dabei kann jeder Client eine eigene Darstellung der Daten vornehmen, da diese von der Anwendung entkoppelt ist. Eine explizite Datenbank garantiert darüber hinaus eine konsistente, effiziente und sichere Verwaltung der anfallenden Daten und einen geregelten Zugriff. So kann auf diese Datenbank auch einfach von externen Anwendungen zugegriffen werden, wie bei der Datenanalyse vorgesehen.

Entwicklungsprozess

Die vorgesehene Entwicklungsmethode entspricht dem vertikalen *Prototyping*. Dabei werden elementweise die einzelnen Teilkomponenten implementiert und diese abschnittsweise auf ihre Funktion getestet. Die notwendigen Grundfunktionen von Datenbank und Webserver müssen dafür zunächst in einem ersten Schritt implementiert werden und können dann nach und nach vervollständigt werden.

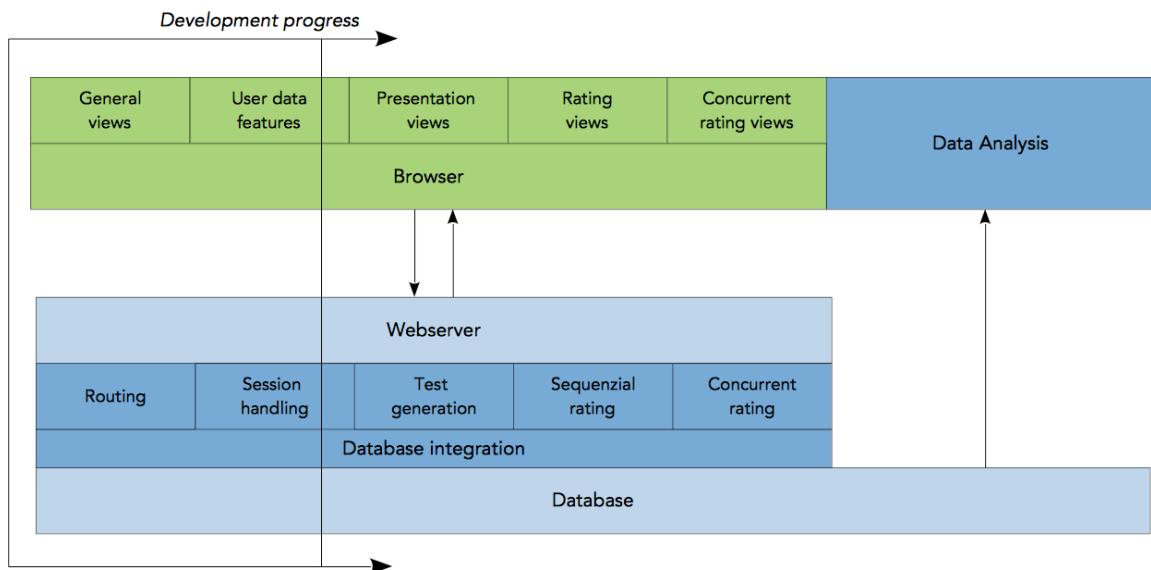


Abbildung 18: Darstellung des Entwicklungsprozesses der einzelnen Komponenten [G6]

4 Implementierung

Im folgenden Abschnitt wird die konkrete Umsetzungen und zentralen Funktionsweisen der benötigten Komponenten beschrieben. Dabei wird auch auf die verwendeten Tools eingegangen, die zur Lösung der einzelnen Problemstellungen eingesetzt wurden.

4.1 Symfony

Die Grundlage der Implementierung stellt das *Symfony 2* Framework dar. Dies ist ein quelloffenes *Web Application Framework*, das seit 2005 mit der Skriptsprache PHP entwickelt wird und aktuell in der Version 2.6 verfügbar ist. *Symfony* bietet Entwicklern von Webanwendungen viel Komfort durch vorkonfigurierte Projekte, eine Vielzahl an Plug-ins und eine übersichtliche Struktur. Dabei bedient sich *Symfony* dem *Model View Controller Pattern*, das für eine strukturiertere Entwicklung von interaktiven Systemen gedacht ist. Dabei wird eine Trennung von Datenmodell (*Model*), Präsentation (*View*) und Programmsteuerung (*Controller*) vorgenommen, um eine größere Flexibilität und Wiederverwendbarkeit zu erreichen. [Sen15]

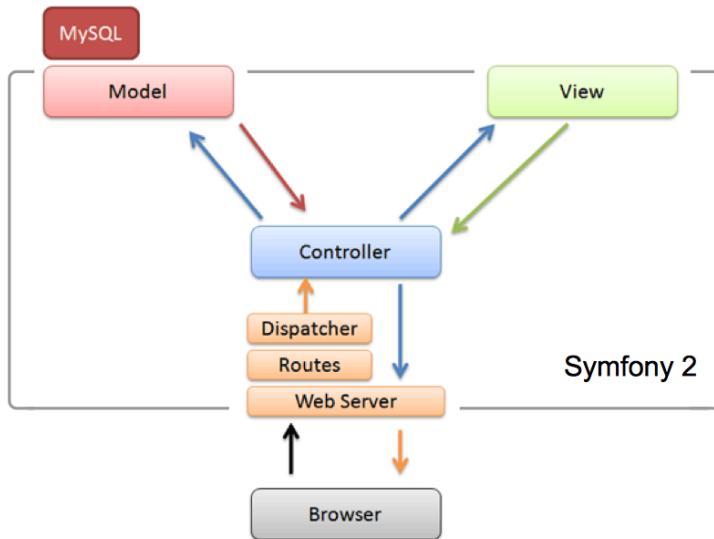


Abbildung 19: Schematische Darstellung des *Modell View Controller* Konzepts für eine Webanwendung [G7]

Wie in Abb. 19 dargestellt, ist der Controller das zentrale Element dieses Entwurfsmusters. Er beantwortet die Anfragen (*HTTP-Requests*) von Benutzern und kann dafür mit View, Datenmodell und dem Nutzer selbst interagieren. In diesem Projekt ist der Controller eine PHP-Applikation, dass die eigentliche Programmsteuerung enthält sowie verschiedenen Algorithmen zur Testgenerierung, Session-Verwaltung und zur Anbindung der Datenbank. Das Datenmodell besteht aus einer MySQL-Datenbank, die durch das Symfony Plug-in *Doctrine* angebunden wird. *Doctrine* ist ein eigenständiges Framework zur objektrelationalen Abbildung von PHP-Objekten in eine relationale Datenbank. Dem Entwickler wird dadurch ermöglicht objektorientiert auf die Daten zuzugreifen und damit auf die Verwendung von SQL Anfragen zu verzichten. So wird nicht nur das Fehlerpotenzial reduziert, sondern auch der Arbeitsaufwand für den Entwickler erheblich minimiert. Die Benutzung von Doctrine bringt außerdem eine verbesserte Performance und Unabhängigkeit vom verwendeten Datenbanksystem.

Der View repräsentiert eine Sammlung von hierarchischen HTML-Dokumenten. Diese wurden für die unterschiedlichen Anwendungsfälle entsprechend der Anforderungen gestaltet. Dabei werden gemeinsame Attribute für die Codereduzierung in einem *baseView* zusammengefasst. Es existieren bspw. *Views* für die Startseite, verschiedene Nutzerdatenabfragen und die unterschiedlichen Testverfahren, wie DCR, ACR oder SAMVIQ. Das MVC-Konzept ist dabei auch deutlich in der Projektstruktur von Symfony wiederzufinden:

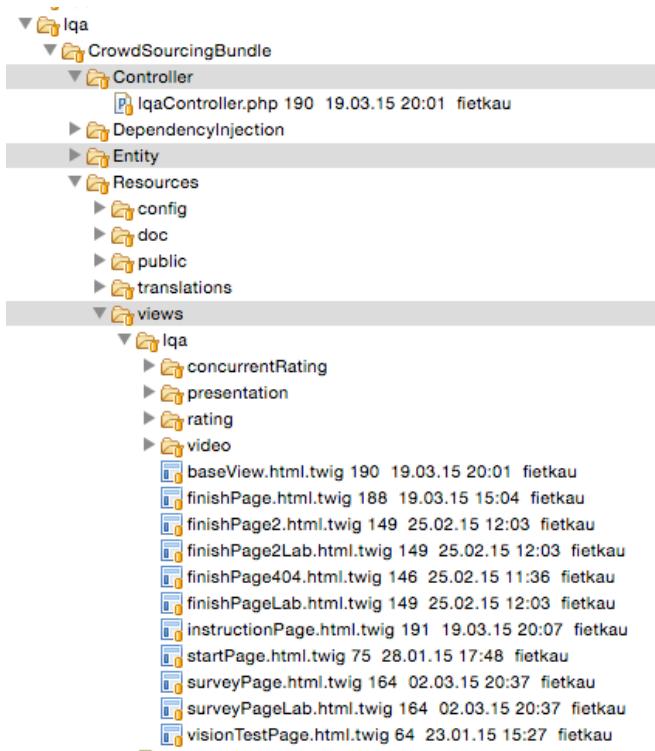


Abbildung 20: Projektstruktur des erstellten Symfony Projekts mit Controller, View und Entity Ordner, der die PHP-Klassen zur Anbindung der Datenbank enthält. [G8]

Ein weiterer Ordner *Web* enthält außerdem die benötigten Bilddaten und andere Dateien für den Webserver, wie CSS-Dateien und die JavaScript-Bibliothek *jQuery*.

4.2 Controller

Der Controller besteht im Wesentlichen aus verschiedenen Methoden, den sogenannten *Actions*. Eine *Action* wird aufgerufen, wenn eine damit verknüpfte Seite über eine URL vom User aufgerufen wird (*Request*). Dafür muss jede *Action* mit einem entsprechenden Aufruf in der routing.yml-Datei registriert werden, wie in Abb. 21 zu sehen:

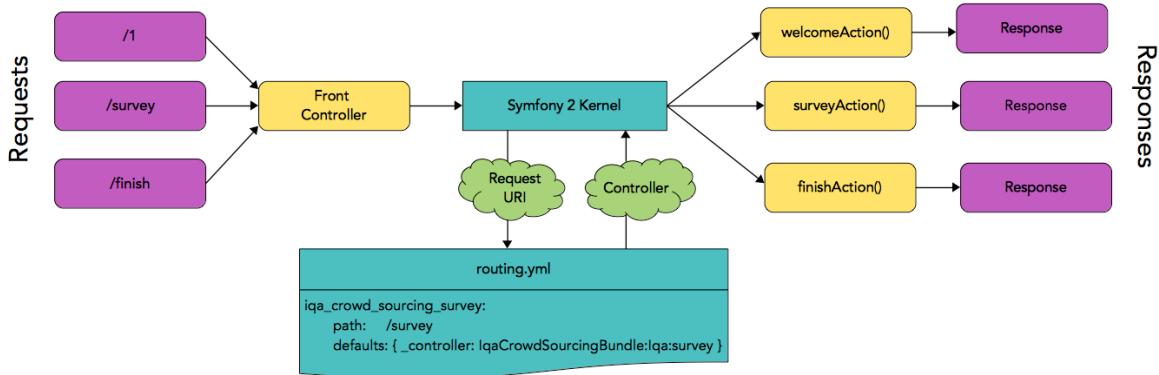


Abbildung 21: URL Routingmechanismus des Symfony Frameworks [G9][Sen15]

Der Großteil der implementierten Anwendungslogik befindet sich dabei innerhalb dieser *Actions*, die Objektmethoden des Controller-Objekts vom *Symfony 2* Framework sind. Für die folgende funktionale Beschreibung dieser Methoden gilt, dass nicht jede *Action* für jedes Testverfahren gleich ist. Vielmehr ist eine sehr feine Verästelung der einzelnen Testverfahren vorzufinden, auf die hier nicht immer detailliert eingegangen wird.

welcomeAction()

Diese Methode wird aufgerufen, wenn ein User die Anwendung im Browser aufruft. Es wird eine neue Session generiert, der Startzeitpunkt abgespeichert und das aktuelle bzw. das in der URL kodierte Experiment geladen. Dass bedeutet, eine Liste mit Stimuli, dazugehörigen Originalen und möglichen *Hidden References* wird für das Assessment aus der Datenbank geholt. Diese Liste wird anschließend für jede Session mit einem Pseudozufallsgenerator, der auf der Grundlage eines linearen Kongruenzgenerators arbeitet, durchgemischt. Anschließend werden die konfigurierten Trainingsstimuli an den Beginn der Liste angefügt. In dem PHP-Objekt *\$session* werden abschließend alle relevanten Informationen gespeichert. Dieses Objekt repräsentiert damit den gesamten Zustand der Applikation. Am Ende der Methode wird die Startseite mit einem Einführungstext über das Experiment gerendert und an den User gesendet.

surveyAction()

Diese Methode liefert eine Formularseite zur Datenerfassung von Nutzerinformationen. Hier werden z.B. für Labor und Web-Assessment unterschiedliche Formulare verwendet. Mögliche Abfragen sind dabei für Alter, Geschlecht, Umgebungshelligkeit, Monitorgröße, Betrachtungsabstand und Expertise des Probanden vorgesehen. Zudem wurden hier auch Methoden implementiert, die eine Voruntersuchungen von Probanden ermöglichen sollen. So können bspw. einige *Pseudoisochromatic Plates* eingeblendet werden und die vom User erkannte Zahl in einem Textfeld abgefragt werden. Nachdem ein User seine Informationen übergeben hat, wird der Input entsprechend geparst und an das Datenmodell übergeben.

instructionAction()

Diese Methode liefert dem User eine Seite mit der jeweiligen Erklärung über das Experiment. Sobald der User seine Kenntnisnahme bestätigt hat, beginnt der Test mit der ersten Sequenz, indem ein *Redirect* auf die erste Testsequenz ausgeführt wird.

presentAction()

Diese Methode unterscheidet sich grundlegend für die einzelnen Testverfahren. Zunächst liefert presentAction() den dazugehörigen und entsprechend gerenderten View der konfigurierten Testmethode. Bei den Verfahren mit sequenzieller Betrachtung und Bewertung, wird nach einem konfigurierten Zeitintervall (presentationDuration) automatisch eine Weiterleitung zur Bewertung (rateAction()) ausgelöst. Handelt es sich aber bei dem Assessment um ein Verfahren bei dem gleichzeitig bewertet und präsentiert wird, empfängt presentAction() hier das vom User gesendete *HTTP-Request* und leitet es dann einfach an rateAction() weiter. Die letzte Funktion von presentAction() ist das detektieren des Testendes. Wurden alle Stimuli vom User bewertet, erfolgt dafür eine Weiterleitung an finishAction().

rateAction()

Diese Methode nimmt mögliche Bewertungen aus *Client-Requests* entgegen und übergibt diese in geeigneter Form an das Datenmodell. Sie ist dabei auch in der Lage eine beliebige Menge an JSON-kodierten Ratings zu empfangen und abzuspeichern, wie es beim SAMVIQ-Verfahren benötigt wird. Sollten keine Bewertungen geliefert werden, liefert rateAction() solange den gerenderten View des verwendeten Bewertungsverfahrens, bis eine Bewertung erfolgt ist.

finishAction()

Diese Methode wird aufgerufen, wenn ein Assessment vollständig von einem User absolviert wurde. Darauf hin wird ein letzter View versendet, mit dem der User sein Feedback hinterlegen kann. Des weiteren wird der Userdatensatz abgeschlossen und ein Zeitstempel für das Testende in die Datenbank geschrieben. Optional ist hier die Möglichkeit einen *Redirect* zum nächsten Experiment einzublenden.

Datenübergabe

Die Übergabe von Eingabedaten erfolgt dabei grundsätzlich nach dem folgenden Prinzip: Der User kann über diverse HTML Elemente (Inputs, Forms, Buttons) ein *HTTP-Request* auslösen. Dieses *Request* entspricht dabei technisch gesehen dem einfachen Anfordern einer Website (*HTTP-GET-Request*). Dabei können allerdings auch beliebige Wertpaare in die URL kodiert werden. Symfony nimmt dieses *Request* entgegen und löst wie bereits beschrieben die definierte *Action* der routing.yml-Datei aus. Mitgelieferte Daten können dann auf der Serverseite mit der Methode get('request') des Controller-Objektes ausgelesen werden. Hat der User Daten gesendet, können diese entsprechend geparsst und weiterverarbeitet werden.

Request ohne Daten:

```
app.php/survey
```

Request mit Daten:

```
app.php/survey?name=bob&age=22&expert=0
```

4.3 Datenbank

MySQL ist ein weitverbreitetes Open-Source-Datenbankmanagementsystem (DBMS) mit einem relationalen Datenbankmodell. Elementarer Bestandteil einer solchen Datenbank sind die Relationen, die wie eine Tabelle über verschiedene Spalten (*Attribute*) verfügen. Eine Zeile in dieser Datenbank repräsentiert dabei einen Datensatz (*Record*). Das DBMS ermöglicht des weiteren mit Hilfe einer *Structured Query Language* (SQL) Relationen zu erstellen, Datensätze zu manipulieren und diese miteinander in Kontext zu setzen. Für die Realisierung der Datenbank wurden damit so die 9 Relationen in Abb. 22 erstellt, die im folgenden näher Erläutert werden.

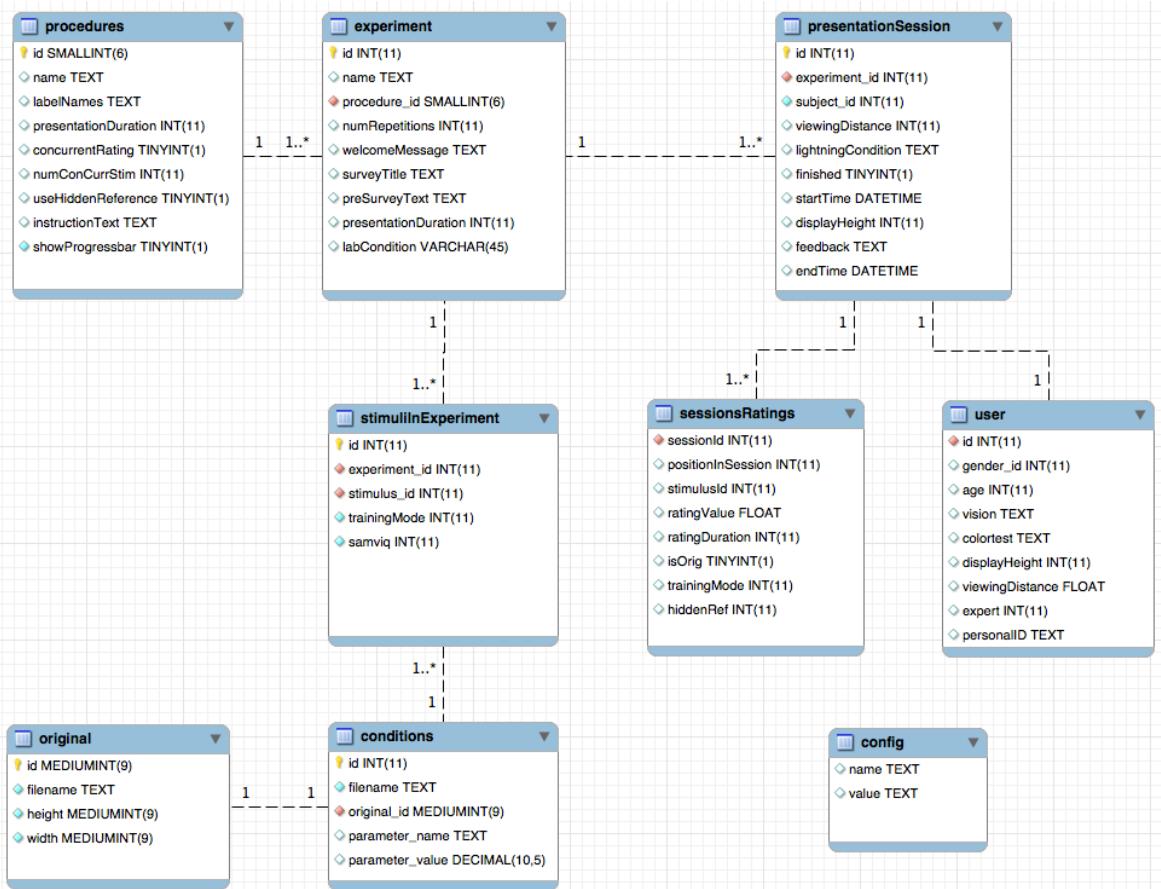


Abbildung 22: Entity-Relationship-Modell des entwickelten Datenbanksystems. [G10]

In der Relation **experiment** werden die jeweiligen Experimente beschrieben. Ein Experiment besteht dabei immer aus einem Namen, hat eine einzigartige ID, eine festgelegte Anzahl an Wiederholungen und ist entweder ein Labor- oder ein Web-Assessment. Für die Reproduzierbarkeit und Fehleranalyse enthält diese Relation außerdem auch diverse Texterklärungen, die im Experiment verwendet werden. Die zu einem Experiment zugeordneten Stimuli (*stimuliInExperiment*), die absolvierten Sitzungen (*presentationSession*) und das zugehörige Testverfahren (*procedure*) werden in einer eigenen Relation gespeichert.

Die Relation **procedures** beschreibt das eigentliche Testverfahren. Es enthält die Erklärung des Testverfahrens (*instructionText*), aber auch eine Beschreibung für den entspre-

chenden Bewertungsmechanismus (*labelNames*). Hier wird außerdem angegeben wie lang die Stimulationszeit ist (*presentationDuration*), ob ein Fortschrittsbalken angezeigt wird (*showProgressBar*), ob versteckte Referenzen verwendet werden (*useHiddenReference*) und ob die Bewertung parallel zur Präsentation stattfinden soll (*concurrentRating*). In der Relation wurden dafür die möglichen Verfahren aus [IT12], [IT08], [IT07a] und verschiedene Variationen davon konfiguriert.

Alle Stimuli die zu einem Experiment gehören, werden in ***stimuliInExperiment*** eingetragen. Diese *Records* sind dabei immer eine Zusammensetzung zweier *Records* der Relationen *original* und *conditions*. Außerdem können Einträge dieser Relation als Trainingsstimuli markiert werden oder für das SAMVIQ-Verfahren einem SAMVIQ-Set zugeteilt werden. Dieses SAMVIQ-Set teilt die Bildern für das SAMVIQ Verfahren in kleine Gruppen ein, die untereinander verglichen werden sollen.

Die Relationen ***original*** und ***conditions*** enthalten jeweils den Dateinamen (*filename*) und die Dimensionen eines Bildes (*width*, *height*). Außerdem speichert *conditions* ein Attribut über den modifizierten Parameter (*parameter_name*) und den konkreten Wert davon (*parameter_value*).

Jedes Experiment hat eine Vielzahl an ***presentationSessions***, die eine Sitzung des jeweiligen Experiments repräsentieren. Eine Sitzung definiert sich dabei durch einen Startzeitpunkt, einen Endzeitpunkt und verschiedene Testbedingungen, wie Umgebungshelligkeit und Abstand vom Monitor. Eine Sitzung entspricht im einfachsten Falle immer auch einem User. Es ist auch darüber hinaus Möglich, einzelnen Usern mehrere Sitzungen zuzuordnen. Dass macht dann Sinn, wenn beispielsweise eine Person mehrere Assessments unter verschiedenen Bedingungen absolviert hat. Daher gehören die entsprechenden Bewertungen der Stimuli immer zu einer Session und nur indirekt zu einem User.

Die Relation ***user*** enthält die Informationen über den Benutzer, wie Alter, Geschlecht, Sehvermögen und Erfahrung im Bereich der Videokodierung. Diese User und ihre Bewertungen können außerdem optional über eine ID (*personalID*) wiedererkannt werden, wenn dies gewünscht ist.

Die Relation ***sessionsRatings*** enthält alle Bewertungen der User aus allen Experimenten. Dafür werden neben dem eigentlichen Wert (*ratingValue*) auch Bewertungsdauer (*ratingDuration*), Sequenzreihenfolge (*positionInSession*) und die mögliche Sonderfunktion der Sequenz (Trainingssequenz, versteckte Referenz, etc.) gespeichert.

Die letzte Relation ***config*** wird lediglich als Konfigurationsdatei benötigt und enthält einige *Key-Value-Paare*, wie ein definiertes *Default-Experiment* und den den Dateipfad zu den Stimuli.

4.4 Präsentation

Die verschiedenen *Views* wurden im Wesentlichen mit HTML und JavaScript erstellt. Für jede Ansicht existiert dafür ein eigenes .html.twig-File. Darin können beliebige HTML und JavaScript Elemente erstellt werden. Die durch Symfony integrierte Funktionalität des Templatings ermöglicht dabei eine Refaktorisierung gemeinsamer Eigenschaften und schafft damit eine Reduzierung von Quellcode und eine zentrale Wartbarkeit. Ein einfaches Beispiel wird im folgenden Listing dargestellt:

```
HTML BasisView Template
```

```
<!DOCTYPE html>
<html>
    <head>
        <meta ... />
        <title>Basis View mit vererbaren Attributen</title>
        <script type="text/javascript">
            alert("Ein Funktionsaufruf im Basis-View den auch Kinder erben");
        </script>
    </head>
    <body class='iqa'>
        {% block body %}
            <!-- Code aus abgeleiteten Views -->
        {% endblock %}
    </body>
</html>
```

```
HTML ChildView: Erbt vom BaseView
```

```
{% extends 'IqaCrowdSourcingBundle:Iqa:baseView.html.twig' %}
{% block body %}
    <!-- Eigenen Funktionen die den BaseView erweitern -->
{% endblock %}
```

Um diese Funktionalität bereitzustellen ist eine Vorverarbeitung der *Views* von Symfony notwendig, die durch einen Renderingmechanismus durchgeführt wird. Eine weitere Funktionalität dieser dynamischen Erstellung ist dabei außerdem die Möglichkeit der Parameterübergabe von *Controller* an den *View*. Dafür werden Wertpaare an die Renderingmethode des *Controllers* übergeben, die im View mit dem {{ Variablename }}-Operator verwendet werden können:

```
PHP Codebeispiel: Parameterübergabe an Views
```

```
return $this->render(
    'IqaCrowdSourcingBundle:Iqa:/concurrentRating/pc.html.twig',
    array( 'filename' => $nextFilename,
           'width'=>$width, 'height'=>$height,
           'usedSliderLabel'=> $sliderLabel,
           'progressBar'=>$session->get('progressBar'),
           'currentImageCount'=>($nextIdsIdx+1))
);
```

Mit dieser Technik wird für jedes Testverfahren daher nur ein View benötigt. Der Algorithmus im Controller steuert dann bspw. über diese Parameter welche Sequenz dargestellt wird. Mit dieser Technik wurden verschiedene *Views* für die Startseite, Nutzerdatenabfragen und die unterschiedlichen Testverfahren entwickelt. Generell erben dafür alle *Views* von einem *baseView*, der gemeinsame Funktionen zu Zeitmessung, Ablaufsteuerung und zur grafischen Gestaltung beinhaltet.

Einen Großteil der Funktionalität eines *Views* wird typischerweise mit JavaScript realisiert. Erst dadurch wird es möglich interaktive Elemente zu entwickeln und bspw. zeitgesteuerte Aktionen durchzuführen. Dabei unterstützen den Entwickler oft komplexe und weit verbreitete *Libarys*, die eine große Menge an Standardfunktionalität mitliefern. Um bspw. einen geeigneten Schieberegler zu integrieren, wurde dafür das *Slider-Objekt* der *Libary jQuery* mit dem Plugin *Pips* verwendet. Die eigentliche Aufgabe reduziert sich so auf die Gestaltung und Skalierung des vorgegebenen Schiebereglers.

4.5 Ergänzendes

Der folgende Abschnitt wird einige weiterführende Erläuterungen über grundlegende Probleme enthalten, die bei einem webbasierten Assessment eine wichtige Rolle spielen.

Zunächst musste sichergestellt werden, dass die angezeigten Stimuli auch in der tatsächlichen Größe angezeigt werden. Denn eine Skalierung des Bildes, wie sie durch die Zoom-Funktion des Browsers und durch eine abweichende Punktdichte der Displays provoziert wird, ist im Falle von Image Quality Assessments störend.

Ein einfaches Workaround für die abweichende Punktdichte der Displays ist das Angleichen der Bildgröße mit dem Attribut *devicePixelRatio* des *Window-Objektes*, das ein Teil der Browser API ist. Dieses Attribut entspricht dabei dem Verhältnis zwischen physikalischen Pixeln und den in HTML verwendeten geräteunabhängigen Pixeln. [Net15]

```
window.devicePixelRatio = physical pixels / logical pixels
```

Stimmen beide Pixelgrößen überein, hat *devicePixelRatio* den Wert 1. Bei einem Retina-Display mit 4-facher Auflösung besitzt *devicePixelRatio* den Wert 4. Für eine korrekte Anpassung der Stimuli wird daher die Bildgröße jeweils mit dem Reziproken des *window.devicePixelRatio* multipliziert.

Die Deaktivierung der Zoom Funktion ist von den Browserherstellern nicht gewollt, daher existiert dafür leider keine einfache Lösung. Eine Option ist die Unterdrückung der entsprechenden Hotkeys (STRG + '+'). Der User kann aber schlussendlich nicht davon abgebracht werden seinen Zoom auf einem anderen Weg zu aktivieren. Dabei würde allerdings die zuvor beschriebene Skalierung mit dem *devicePixelRatio* immer dafür sorgen, dass die nächste Sequenz wieder korrekt angezeigt wird. Letztlich bleibt daher nur die Möglichkeit zum Beginn des Assessments die User darauf aufmerksam zu machen keinen Zoom zu verwenden.

Ein weiteres Problem ergibt sich durch den mangelnden *Quality of Service* im Web. Im Falle von geringer Bandbreite oder schlechter Latenz kann es dazu kommen, dass die Testsequenzen zu langsam laden oder der Bildaufbau sichtbar wird. Im Falle eines Labor-experiments kann dies zunächst durch eine gute Netzwerkanbindung vermieden werden. Ein weiterer Schritt ist die implementierte Vorladetechnik. Dafür werden beim Aufrufen der Anweisungsseite bereits alle URLs der im Assessment benötigten Bilder übergeben. Eine kleine JavaScript Funktion sorgt dann im Hintergrund dafür, dass alle Bilder in den *Browser-Cache* geladen werden. Wird ein Bild im späteren Verlauf des Assessments benötigt, kann dieses sehr viel schneller aus dem lokalen *Cache* geholt werden und muss

nicht erst vom Server angefordert werden. Für die Videos ist dieses Vorgehen, durch die geringen Cachegrößen so leider nicht machbar. HTML5 sieht allerdings ein Preload-Tag für Video-Elememte vor. Dieser sorgt dafür, dass eine Videosequenz sofort beim Aufruf der Seite geladen wird, was immerhin eine kleine Verbesserung bewirkt.

Des weiteren wurden noch einige Ideen umgesetzt, die aus Probeversuchen mit der Software stammen. Um den Teilnehmer ein visuelles Feedback ihres Fortschrittes zu geben und ihre Motivation zu halten, kann ein Fortschrittsbalken eingeblendet werden. Außerdem konnte beobachtet werden, dass ein Proband bei Verfahren mit begrenzter Stimulationszeit manchmal keine Entscheidung treffen konnte. Das kann vielfältige Gründe haben, wie bspw. mangelnde Konzentration oder zu schwere Fragestellungen. Um den Teilnehmer in dieser Situation nicht zu einer Bewertung zu zwingen, wurde ein *Skip-Mechanismus* implementiert. Damit dabei nicht das Testverfahren ohne Bewertung verlassen werden kann, wird die übersprungene Sequenz immer am Testende eingefügt.

4.6 Datenanalyse

Die Auswertungsskripte wurden mit der Programmiersprache Python erstellt. Die verwendeten Bibliotheken *matplotlib* und *pandas* liefern die benötigte Datenstrukturen und eine umfangreiche Funktionalität zur Auswertung und Visualisierung von Messdaten.

Für den einmaligen Zugriff auf die Datenbank wird die Bibliothek *MySQLdb* genutzt. Eine komplexe SQL-Anfrage verschmilzt dafür die verschiedenen Datenbankrelationen zu einer allumfassenden Relation mit den benötigten Ratings. Das Resultat dieser Abfrage wird dann in einem sogenannten *pandas.DataFrame*-Objekt gespeichert. Diese Datenstruktur liefert alle benötigten Funktionen zur Auswertung der Daten, wie filtern, gruppieren und umfangreiche statistische Funktionen.

Um eine übersichtliche Präsentation der Auswertungsergebnisse zu erlangen, wird ein *IPython Notebook* verwendet. Dies ist eine webbasierte, interaktive Arbeitsumgebung in der Code, Text, Formeln, Grafiken und andere Multimediaelemente in einem Dokument kombiniert werden können. Mit dieser Umgebung wurden so die in [IT12] und [IT08] empfohlenen Funktionen sowie viele weitere darüber hinaus entwickelt.

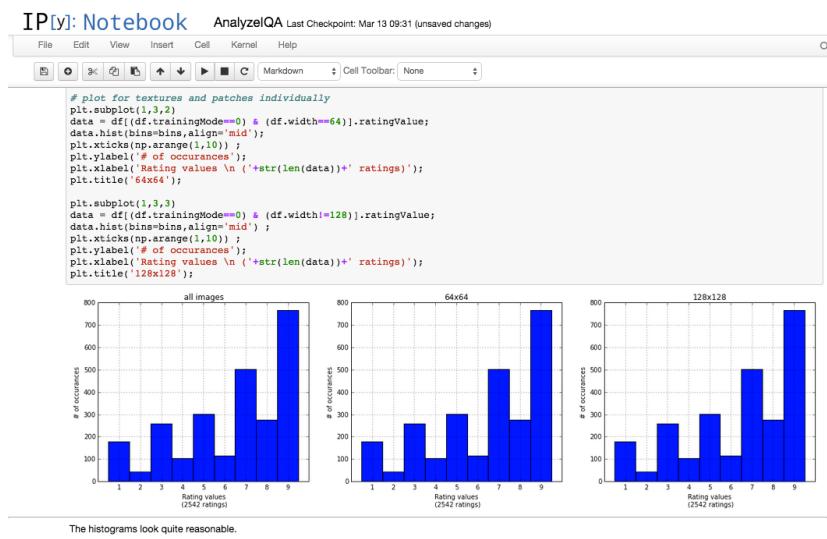


Abbildung 23: Screenshot eines IPyhton Notebooks [G11]

4.7 Setup

Als Voraussetzung für den Start der Anwendung werden PHP, Composer und MySQL benötigt. Die verwendete Ausführungsplattform für die Entwicklung ist Ubuntu/Linux. Generell ist das Projekt durch Symfony aber Plattformunabhängig. Die benötigten *Dependencies* müssen dafür einmalig mit der PHP-Paketverwaltung Composer installiert und eine MySQL Datenbank konfiguriert werden. Alle benötigten Komponenten befinden sich auf der CD im Anhang.

Installation aller *Dependencies* im Projektverzeichnis:

```
$ composer require symfony/finder
```

Eventuell ist ein Start des MySQL Servers notwendig:

```
$ mysql.server start
```

Auf jeden Fall muss aber ein Backup der MySQL Datenbank eingespielt werden:

```
$ mysql -u {username} -p{password} iqa < {database_backup_file}
```

Das Bootstraping der eigentlichen Symfony Anwendung erfolgt durch den Start des PHP Servers aus dem Projektverzeichnis heraus:

```
$ php app/console server:run {ip}:{port}
```

Anschließend kann die Anwendung im Browser aufgerufen werden.

```
http://localhost:8000/app.php/11
```

Zum testen der Anwendung wurden die verschiedenen Verfahren als Beispielexperimente konfiguriert. Diese können dafür nach einem erfolgreichen Anwendungsstart auf den URLs von /11 bis /35 betrachtet werden.

Für die Verwendung der Datenanalyse muss außerdem der IPython Notebook Server gestartet werden:

```
$ ipython notebook
```

Dieser Dienst kann mit dem Browser auf <http://localhost:8888/tree> erreicht werden. Dort kann ein entsprechendes Notebook geöffnet und ausgeführt werden.

5 Laboreinrichtung

5.1 Ausstattung

Ein speziell für *Image Quality Assessments* eingerichteter Raum am HHI liefert für das Vorhaben bereits die Grundlagen, wie einstellbare Umgebungshelligkeit, neutraler Anstrich und benötigte Hardware. Um einen geeigneten Labortest unter den Bedingungen der ITU durchführen zu können, werden noch diverse Einstellungen und Messungen benötigt. Im folgenden werden dafür zunächst die einzelnen Schritte für die Konfiguration des Testplatzes erläutert. Abschließend wird dann eine erarbeitet Checkliste einen Vergleich zwischen den konfigurierten und den vorgeschriebenen Werten ermöglichen.

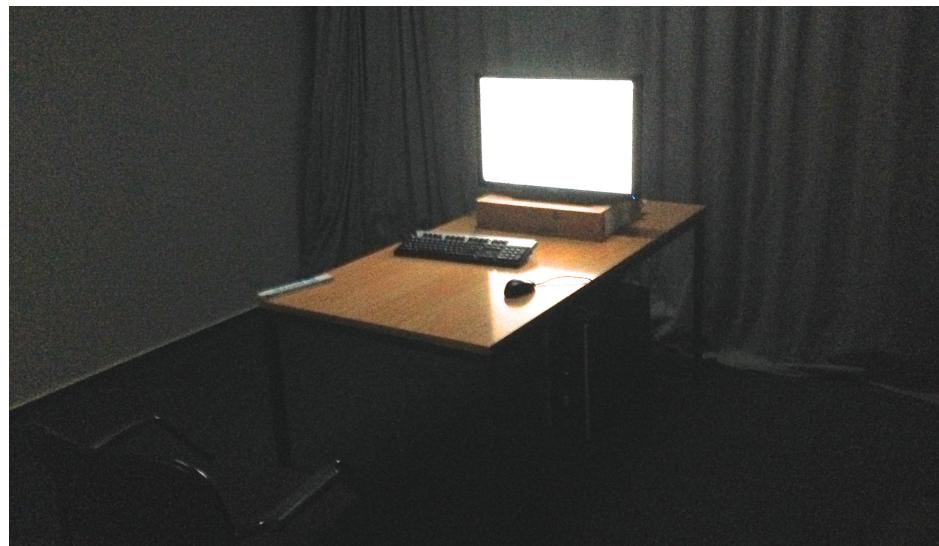


Abbildung 24: Konfigurierter Testplatz mit geringer Umgebungshelligkeit [G12]

Umgebungshelligkeit

Da der Raum über keine Fenster verfügt, wird die Umgebungshelligkeit ausschließlich durch eine Leuchtstoffröhre, unmittelbar an der Decke über dem Testplatz, erzeugt. Um eine geringe Umgebungshelligkeit während des Assessments zu realisieren, wird dafür der Dimmer mit minimaler Helligkeitseinstellung betrieben. Durch die Graue Farbe des Raums wird außerdem kaum Licht reflektiert.

Monitor

Der verwendete Monitor ist ein 27-Zoll LCD-Gerät der Marke Dell vom Typ U2711b. Er hat eine native Auflösung von 2560 x 1440 Pixeln (16:9) und hat eine nutzbare Bildfläche von 597 x 336 mm. Dabei beträgt die Größe eines Pixels (Pixelpitch) 0.233 mm.

Sonstiges

Für ein Assessment steht im Labor ein PC mit dem Betriebssystem Windows und dem Browser Chrome zur Verfügung. Chrome kann dafür zum Beginn eines Versuchs in den Vollbildmodus geschalten, wodurch nur noch der Inhalt einer Website zu sehen ist. Dem Probanden stehen zur Bedienung der Applikation Maus und Tastatur zur Verfügung.

5.2 Messungen

Für die Überprüfung, der durch die ITU-Richtlinien vorgegebenen Parameter, wurden zwei Messgeräte verwendet. Dafür wurde zunächst eine Kalibrierung des Monitor vorgenommen und anschließend eine Helligkeitsmessungen vorgenommen. Um eine gut abgestimmte Konfiguration zu erreichen, ist ein mehrfaches Durchlaufen und Nachbessern der Parameter notwendig gewesen. Die endgültige Konfiguration stellt sich wie folgt dar:

Vorbedingungen:

1. Monitor vorwärmen (30 min)
2. Geringe Umgebungshelligkeit (Minimale Dimmereinstellung)

Monitor Kalibrierung

Für die Monitor-Kalibrierung wurde das Messgerät *Spyder 3* der Firma Datacolor verwendet. Dieses besteht aus einer Software und einer optischen Messeinheit, die während der Kalibrierung direkt auf dem Bildschirm platziert wird. Über das USB-Interface übermittelt diese Messeinheit diverse Messwerte an die Software. So kann automatisiert die beste Übereinstimmung mit der gegebenen Konfiguration gesucht werden. Abschließend erstellt die Software ein entsprechendes Protokoll mit einer Übersicht der kalibrierten Parameter. Für die Kalibrierung wurden folgende Arbeitsschritte angewandt:

1. Factory Reset des Monitors und Konfiguration der Helligkeit auf Stufe 70
2. Starten des Spyder Software zur Kalibrierung mit den Einstellungen:

Konfigurationsvorlage: ITU Rec. 760

Gamma : 2.2; Farbtemperatur 6504 K; Helligkeit: 200 cd/m²

3. Start der Kalibrierung

Das automatisch generierte Kalibrierungsprotokoll ist in Abb. 25 zu finden. Eine weitere Messung zur Bestimmung der Umgebungshelligkeit lieferte das Resultat: *low - very low*

Generic PnP Monitor-1		
Luminanz (Candelas):		
	Schwarz	Weiß
Unkalibriert	0,36	228,6
Zielwert	0,36	200,0
Kalibriert	0,36	197,9
Weißpunkt (CIE xy):		
	Weißenpunkt	
Unkalibriert	0,313	0,337
Zielwert	0,313	0,329
Kalibriert	0,311	0,328
Phosphor (CIE xy):		
	Rot	0,670
	Grün	0,181
	Blau	0,152
		0,315
		0,669
		0,043
DeltaE (Lab):		
	Weißenpunkt	1,4
	50% Grau	0,2
Gamma:		
	Unkalibriert	2,01 (0,05)
	Zielwert	2,20 (0,00)
	Kalibriert	2,21 (0,01)

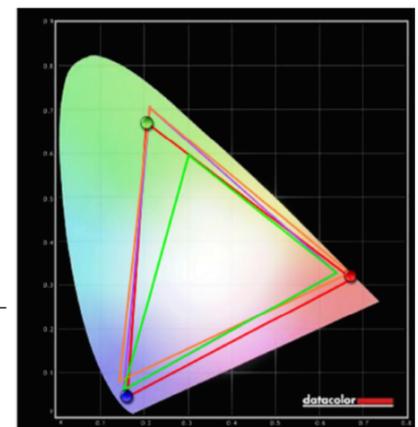


Abbildung 25: Das Kalibrierungsprotokoll der *Spyder 3* Software mit Vergleich im CIE-Normalenzsystem von AdobeRGB (Lila), sRGB (grün), NTSC (orange) und der Zielkonfiguration (rot) [G13]

Helligkeitsmessung

Für die Helligkeitsmessung wurde das Gerät CS-100A der Firma Konica Minolta mit folgender Konfiguration verwendet. Zunächst wurde das Gerät in den Auslieferungszustand zurückgesetzt. Anschließend wurden die Standardkonfiguration nach Handbuch eingestellt: Preset; ABS Mode; Fast Response und manuelle Adjustierung. Für die Helligkeitsmessung wurden außerdem 3 unterschiedliche Messpunkte bestimmt. Unter Verwendung der gemessenen Werte, konnte so anschließend die erstellte Checkliste bearbeitet werden.

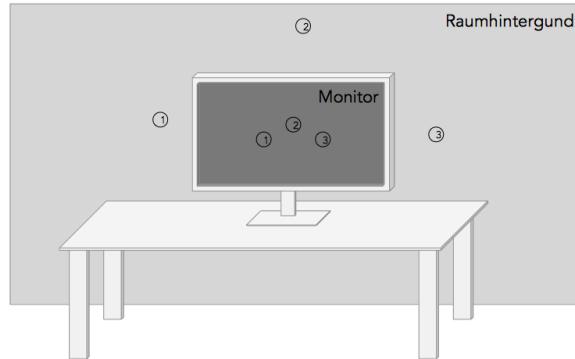


Abbildung 26: Messpunkte für Bildschirm- und Raumhelligkeit mit CS-100A [G14]

cd/m^2	Abstand	Messung 1	Messung 2	Messung 3	Mittelwert
Monitor - schwarz	1 m	0.24	0.22	0.20	0.22
Monitor - weiß	1 m	190	188	172	183.3
Monitor - inaktiv	1 m	-	0.01	-	0.01
Raumhintergrund	2 m	0.39	0.43	0.38	0.40

ITU Richtlinien (soll)	Tatsächlicher Wert (ist)
Maximale Helligkeit des Bildschirms $\approx 200 cd/m^2$	$190.6 cd/m^2$ Mittelwert von CS100A (183.3) und Spyder3 (197.9)
Helligkeitsverhältnis von schwarzer zu weißer Farbdarstellung bei dunkler Umgebung ≤ 0.1	$0.22 / 183.33 = 0.0011$ (CS100A) $0.36 / 197.9 = 0.0018$ (Spyder3) [25]
Helligkeitsverhältnis zw. Raumhintergrund und max. Bildhelligkeit ≤ 0.2	$0.40 / 183.33 = 0.0021$
Helligkeit und Kontrast setup via PLUGE	Spyder3 nach ITU Rec 709 Helligkeit, Kontrast, Weißpunkt
Maximaler Betrachtungswinkel $< 30^\circ$	Frontal
Abstand des Betrachters nach PVD oder je nach Versuchsziel:	Frei Konfigurierbar
Minimale Bildschirmdiagonale 14 Zoll	Bildschirmdiagonale = 27 Zoll
Ungenutzte Bildschirmflächen mit 50 % grau gefüllt ($Y=U=V=128$)	HTML Hintergrundfarbe #808080

6 Experiment

Die implementierte Anwendung sowie die Laborumgebung wurde anschließend in einem Assessment erprobt. Dabei sollten die folgenden Fragestellungen beantwortet werden:

1. Aus der Motivation und den erarbeiteten Grundlagen geht klar hervor, wie stark der Unterschied zwischen der menschlichen Wahrnehmung und einigen Metriken ist. Anknüpfend daran soll daher die Korrelation zwischen MOS und den Metriken PSNR sowie SSIM untersucht werden.
2. Eine aktuelle Entwicklung im Bereich der IQAs sind *crowdbasierte* Assessments, deren vielfältige Eigenschaften bereits diskutiert wurden. Dabei unterliegt bekanntermaßen jeder Einzeltest anderen Bedingungen, was somit zu schlechteren Ergebnissen führen sollte. Um diese Umstände näher zu untersuchen, werden daher identische Web- und Laborexperimenten gestartet. Hiermit soll untersucht werden, wie stark sich die Bewertungen der *Crowd* von denen aus dem Labor unterscheiden.
3. Eine letzte Fragestellung hat sich aus aktuellen Arbeiten am Fraunhofer HHI ergeben. Bei unterschiedlich großen Bildausschnitten mit gleicher Qualität, sollten größere Stimuli auch eine bessere Qualitätsbewertung ermöglichen. Eine zentrale Frage dabei ist, ab welcher Größe diese verbesserte Wahrnehmung eintritt und wann ein Proband mögliche Störungen aussagekräftig beurteilen kann? Um diesen Sachverhalt zu untersuchen sollen die Bewertungen von kleineren Versionen der Stimuli (64x64) mit größeren Versionen (128x128) verglichen werden.

6.1 Konfiguration

Um die gegebene Fragestellung untersuchen zu können, wurden 4 unterschiedliche Assessments in der Datenbank konfiguriert.

...

Das angewandte Verfahren ist das *Double Stimulus* Verfahren (DCR) mit der 9-stufigen ITU-Impairment Skala. Die Präsentation von Referenz und Testsequenz wird gleichzeitig durchgeführt, da es sich um sehr kleine Stimuli handelt. Die Stimuli dürfen dafür so lange betrachtet werden wie gewünscht und die Bewertung kann nach einer beliebigen Zeit abgeben werden. Die Reihenfolge der konfigurierten Stimuli ist in dem Assessment für jeden Probanden zufällig. Die maximale Gesamtdauer des Tests sollte 30 min nicht übersteigen. Im Labor kann dies durch den Versuchsleiter sichergestellt werden, während im Web-Assessment nur eine nachträglich Auswertung der Zeitstempel möglich ist.

Stimuli

Jedes Experiment mit gleicher Stimuligröße enthält für den Versuch auch die gleichen Stimuli. Diese wurden dafür mit abweichendem Qualitätsparameter (QP) nach dem *High Efficiency Video Coding* (HEVC) Standard kodiert. Der QP steht in einem direkten Zusammenhang mit der Bildqualität und kann einen Wert im Intervall [0, 51] annehmen [VS14]. Die 24 homogenen Texturen die eine gemeinsame Schnittmenge in allen Experimenten darstellen, können der folgenden Übersicht entnommen werden:

Dateiname	QPs
blanket1-a_lumAdjusted.png	31, 39, 43, 51
gray_flakes001-inca-120dpi.png	31, 38, 43, 51
gray_rubber001-inca-300dpi.png	25, 31, 36, 44
oatmeal1-a_lumAdjusted.png	32, 38, 43, 51
scarf1-a_lumAdjusted.png	32, 44, 50, 51
stone1-a_lumAdjusted.png	32, 39, 43, 50

Die dafür notwendige Größenanpassung in 128x128 und 64x64 große Versionen der Bilder, wurde mit der Python Bibliothek PIL wie folgt durchgeführt:

```
>> Image.open({filename}).crop(0, 0, 64, 64)
>> Image.open({filename}).crop(0, 0, 128, 128)
```

Darüber hinaus sind in allen Experimenten mit gleicher Bildgröße noch weitere Stimuli enthalten. Dabei handelt es sich um Ausschnitte (*Patches*) aus natürlichen Videosequenzen die häufig als Testsequenzen verwendet werden. Auch diese Originalsequenzen wurden mit je 4 verschiedenen Qualitätsparametern (QPs) kodiert was somit 48 weitere Stimuli der Größe 64x64 und 128x128 ergibt. Eine vollständige Übersicht aller verwendeten Stimuli befindet sich im Anhang 1. Die somit insgesamt 72 Stimuli je Test werden zwei mal Wiederholt und enthalten zusätzlich 5 weitere Trainingsstimuli. Jedes konfigurierte Assessment besteht somit aus insgesamt 149 Bewertungen.



Abbildung 27: Beispiele für Texturen der Größe 64x64 mit Originalsequenz (links) und den verschiedenen Testsequenzen (von links nach rechts) mit QP 31, 39, 43 und 51. [G15]

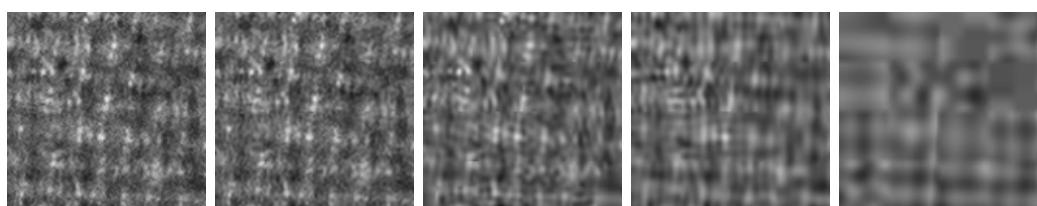


Abbildung 28: Beispiele für Texturen der Größe 128x128 mit Originalsequenz (links) und den verschiedenen Testsequenzen (von links nach rechts) mit QP 31, 39, 43 und 51. [G16]

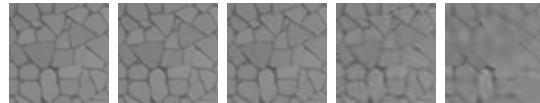


Abbildung 29: Beispiele für *Patches* der Größe 64x64 mit Originalsequenz (links) und den verschiedenen Testsequenzen (von links nach rechts) mit QP 21, 25, 31 und 36. [G17]

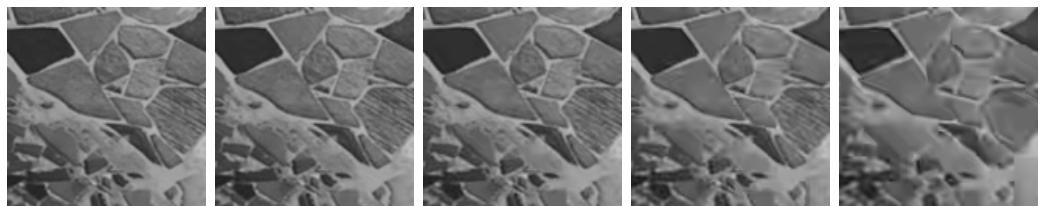


Abbildung 30: Beispiele für *Patches* der Größe 128x128 mit Originalsequenz (links) und den verschiedenen Testsequenzen (von links nach rechts) mit QP 22, 28, 34 und 40. [G18]

Betrachtungsabstand

Für das Vorhaben sehr kleine Bildausschnitte zu testen, ist leider keine konkrete Aussage in den ITU-Richtlinien zu finden. Da die Testsequenzen aus Sequenzen der Größe 1920x1080 extrahiert wurden, wird die Berechnung des Betrachtungsabstand auf Grundlage dieser Originalgröße getroffen:

$$\text{Bildhöhe} \cdot \text{Pixelgröße} = 1080 \text{ Pixel} \cdot 0.0233\text{cm} = 25.16\text{cm}$$

Da diese Bildhöhe der tatsächlichen physikalischen Bildhöhe der Originalsequenz auf dem Monitor entspricht, wird diese auch als Bezugswert verwendet. Bei einer Bildschirmgröße ab 24“ und einer Bildhöhe zwischen 23 bis 30 cm empfiehlt die ITU, nach PVD-Tabelle in [IT12], einen Betrachtungsabstand zwischen 7 und 8 H. Dies entspricht einem Abstand von 1.76 bis 2.01 Metern. Allerdings widersprechen [IT08] und [IT07a] diesem Vorgehen und besagen, dass der Betrachtungsabstand von der jeweiligen Anwendung und dem Ziel des Experiments abhängig gemacht werden sollte. Zudem ergibt sich das Problem, dass der Abstand für das äquivalente Web-Assessment nicht kontrolliert werden kann. Hier musste die Annahme getroffen werden, dass der typische Abstand eines Computerarbeitsplatzes eingenommen wird. Für die beste Vergleichbarkeit zwischen Web- und Laboruntersuchung, wurde daher letztlich der Abstand von 4 H bestimmt, was ungefähr einem Meter entspricht ($4 \cdot 25.16 \text{ cm} = 100.64 \text{ cm}$). Um diesen Betrachtungsabstand zu realisieren, werden alle Probanden zum Testbeginn korrekt platziert und dazu angewiesen ihren Abstand bestmöglich beizubehalten. Als Vergleichspunkt dient hierbei der konfigurierte Abstand von einem Meter zwischen Tischkante und Monitor, wie auch in Abb. 24 zu sehen.

6.2 Durchführung

Die Durchführung der Laborexperimente fand im Zeitraum vom 23.02.2015 bis 27.02.2015 statt. Dafür wurden die Probanden individuell zum Test eingeladen. Das alternative Web-Assessment wurde parallel dazu gestartet und vorwiegend in einem vergleichbaren Zeitraum von den Teilnehmern absolviert. In allen Experimenten besteht der überwiegende Teil der Probanden aus Mitarbeitern der Abteilung *Image Processing* des Fraunhofer Heinrich Hertz Instituts. Darunter befinden sich auch Probanden die keine Erfahrung im Bereich der Bild- und Videokodierung haben. Auf eine vorherige Erklärung an Beispielen wurde verzichtet. Jeder Test startet allerdings mit 5 Teststimuli die in etwa das gesamte Spektrum der vorkommenden Qualitätsstufen umfassen. Diese werden in der anschließenden Auswertung nicht berücksichtigt.

6.3 Auswertung

In den Assessments konnten dabei 10.299 gültige Bewertungen und 421 Trainingsbewertungen gesammelt werden. Mithilfe der Python-Skripte im *IPython Notebook* WebvsLab-Final wurde für die Experimente die folgenden Auswertungen erstellt:

	Lab64	Lab128	Web 64	Web 128	Einheit
Bewertungen	2542	2448	3414	1895	
Sitzungen	18	17	32	14	
Vollständige Sitzungen	17	17	20	13	
Experten	15	14	18	6	
Mittlere Dauer	11.74	10.44	16.25	28.53	Minuten
Standardabw. Dauer	2.76	3.10	11.40	52.56	Minuten
Mittlere Dauer je Stimuli	4.5	3.9	7.7	10.9	Sekunden
Mean Opinion Score	6.39	5.94	5.58	5.24	ITU 9er Skala
Durchschnittsalter	29.21	28.88	34.22	36.49	Jahre

Viele Punkte dieser Auswertung unterstützen die zuvor Diskutierten Vor- und Nachteile der crowdbasierten Assessments. Am Web-Assessment haben 46 Probanden und im Laborexperiment 35 teilgenommen. Der Durchführungsaufwand im Labor war sehr hoch, da die Probanden jeder für sich ermutigt und eingeladen werden mussten und zudem die Prozedur nur sequenziell durchlaufen konnten. Alternativ reichte für das *crowdbasierte*-Assessment eine Einladung via E-Mail, um eine viel bessere Teilnehmerzahl zu erreichen. Die ungleich verteilte Probandenmenge im Web kann dadurch erklärt werden, dass bei Vielen nach dem ersten Test keine Lust auf einen weiteren vorhanden war. Die sinkende Motivation für Web-Assessments spiegelt sich dabei auch in den Testabbrüchen wieder. Während im Labor nur ein einziger Abbruch vorhanden war, weil der Proband in 28 Minuten nur 94 von 288 Stimuli bewertet hatte, sind im Web rund 30% der Sitzungen abgebrochen worden. Außerdem ist dort ein signifikanter Anstieg in der Bearbeitungsdauer zu beobachten. Diese Entwicklung kann viele Gründe haben: Einerseits sind in manchen Sitzungen lange Pausen zu beobachten, andererseits scheinen einige Probanden auch dazu bereit zu sein mehr Zeit zu investieren.

Screening

Das Screening nach ITU Richtlinie BT.500 wurde für jedes Experiment separat durchgeführt. In der folgenden Übersicht werden nur Probanden aufgeführt für die mindestens eine der beiden Ausschlussbedingungen zutrifft. Die erste Bedingung prüft dafür ob mehr als 5% der Bewertungen eines Probanden außerhalb der doppelten Standardabweichung liegen bzw. bei nicht normalverteilten Bewertungen außerhalb der $\sqrt{20}$ -fachen Standardabweichung. Die zweite Bedingung prüft unterdessen auf eine ausgewogene Verteilung der Fehlbewertungen. Erst wenn ein Proband im entsprechenden Maße über- sowie unterhalb der erlaubten Abweichung liegt trifft diese Bedingung zu.

Sitzung	P	Q	L	$(P + Q) / (L) \geq 0.05$	$ (P - Q)/(P + Q) < 0.3$	Reject
Lab 64						
71	9	0	144	0.0625	false	no
110	0	10	144	0.0694	false	no
57	0	8	144	0.0556	false	no
Lab 128						
102	0	11	144	0.0764	false	no
72	22	0	144	0.1528	false	no
Web 64						
129	2	2	144	false	0.0	no
137	1	1	144	false	0.0	no
150	0	13	144	0.0903	false	no
56	20	0	144	0.1389	false	no
73	6	0	64	0.0938	false	no
81	13	0	144	0.0903	false	no
122	0	10	144	0.0694	false	no
123	0	17	144	0.1181	false	no
Web 128						
128	3	2	144	false	0.2	no
143	2	9	144	0.0764	false	no
112	4	3	144	false	0.143	no
119	1	1	144	false	0.0	no
60	19	0	144	0.1319	false	no

Mit der verwendeten Screening-Methode sind keine User gefunden worden die abgelehnt werden sollten. Allerdings kann hier ein klarer quantitativer Unterschied zwischen Labor und Web aufgezeigt werden: 5 von 35 Probanden ($\approx 14\%$) im Labor haben eine der 2 Bedingungen erfüllt. Im Gegensatz dazu sind 13 von 46 Teilnehmern ($\approx 28\%$) auffällig geworden, was eine Verdopplung der Werten aus dem Labor entspricht.

Erwähnenswert ist an dieser Stelle außerdem, dass die zweite Bedingungen einen Probanden, der immer die beste oder schlechteste Bewertung abgibt, vor einem Ausschluss bewahren würde, was ein klares Problem darstellt.

Vergleich von Mean Opinion Score und Metriken

Eine weitere Analyse der Daten soll den Zusammenhang zwischen *Mean Opinion Score* und den vorgestellten Metriken, wie in [LJ11], untersuchen. Dafür wurden die Streudiagramme in Abb. 31 erstellt.

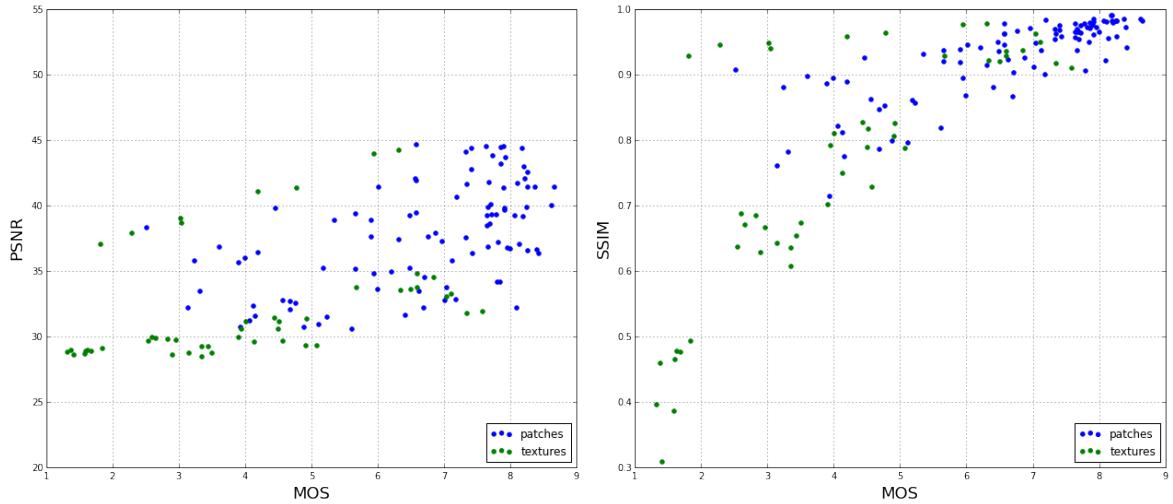


Abbildung 31: Korrelationsbetrachtung zw. MOS und PSNR (links) bzw. MOS und SSIM (rechts) [G19]

Die erforderliche Berechnung des PSNR wird durch folgende Rechenvorschrift in einem Python Skript durchgeführt:

```
>> mse = ((image1 - image2)**2).mean()
>> psnr = 10*np.log10((255**2)/ mse)
```

Der außerdem benötigte SSIM Wert wird durch das Programm *pyssim* berechnet, dass auf Basis von [WaBSS04] und [WB09] entwickelt wurde und unter folgender URL frei zur Verfügung steht: <https://github.com/jterrace/pyssim>

In dem Streudiagramm (Abb. 31) ist klar zu erkennen, dass kein vertrauenswürdiger Zusammenhang zwischen den Ausgaben der Metrik und dem ermittelten MOS-Wert vorhanden ist. Die Korrelation nach Pearson liegt außerdem für die unterschiedlichen Experimente im Mittel bei 0.5356 für den PSNR und bei 0.7918 für den SSIM. Diese Resultate entsprechen den Werten in Abb. 16 aus [LJ11] und bestätigen damit den in [DY09], [WB09] und [LJ11] dargestellten Zusammenhang, der unzureichenden Korrelation zwischen subjektiven und objektiven Bewertungen.

Pearson Korrelation	PSNR	SSIM
Lab64	0.48249	0.77574
Lab128	0.53346	0.76237
Web64	0.57116	0.83235
Web128	0.55545	0.79704
Mean	0.53564	0.79188

Vergleich von Web- und Laborbewertungen

Für den Vergleich von Web und Laborexperiment wurden die Grafiken 32, 33, und 34 erstellt. In Abb. 32 wurden die durchschnittlichen Bewertungen der einzelnen Qualitätsstufen der Originalbilder zusammengetragen, um den Zusammenhang zwischen Qualitätsparameter und *Mean Opinion Score* aufzuzeigen. Hierbei ist deutlich zu sehen, dass Laborexperimente fast immer eine bessere Bewertung der Stimuli liefern.

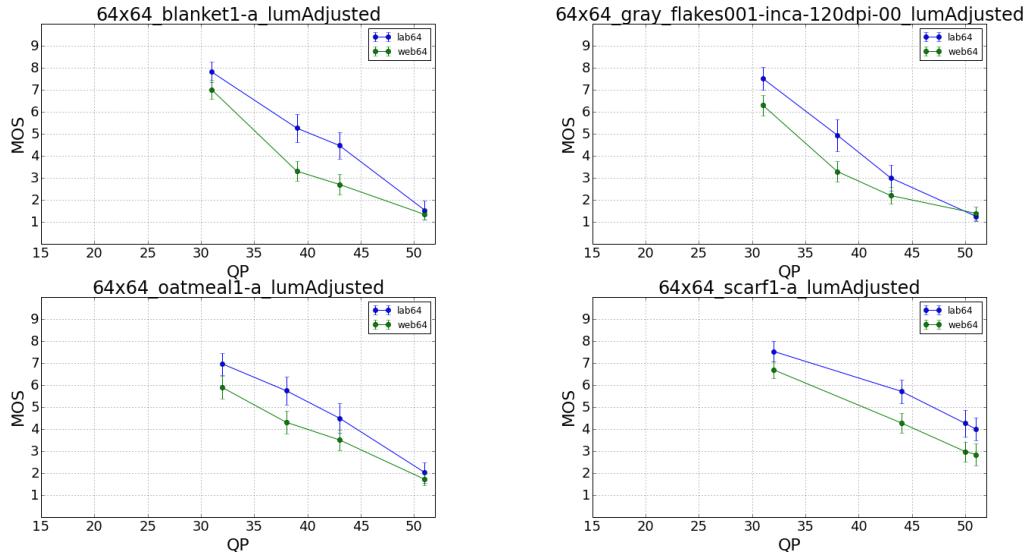


Abbildung 32: Teilbild von der Analyse des Zusammenhangs zw. MOS und QP mit Angabe des Konfidenzintervalls (95%). Eine vollständige Grafik befindet sich im Anhang. [G20]

Wie gut dabei die Ergebnisse aus dem Web sind, zeigt sich in dem Streudiagramm in Abb. 33. Für die Experimente mit gleicher Bildgröße liegt ein guter linearer Zusammenhang vor, wie auch an den Werten der Pearson-Korrelation zu sehen ist. Bei Assessments mit Stimuli der Größe 64x64 Pixeln beträgt diese 0.964010 und bei den größeren Stimuli 0.968371. Die leichte Neigung in Abb. 33 bestätigt, die bereits zuvor in Abb. 32 erkannte Tendenz zu einer schlechteren Qualitätsbewertung der Stimuli im Web.

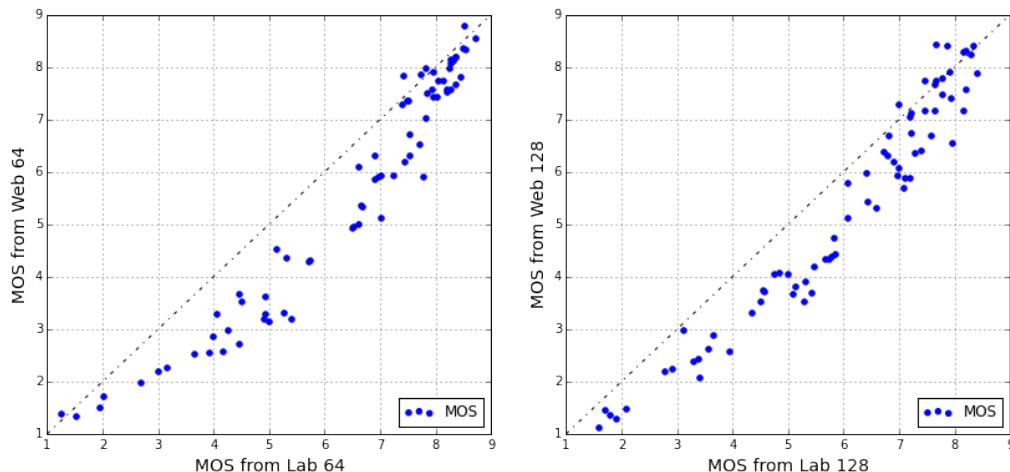


Abbildung 33: Streudiagramm zum Vergleich der Bewertungen aus Labor und Web [G21]

Für die Untersuchung der Aussagekraft von Labor und Web-Assessment wurde außerdem eine Auswertung des Konfidenzintervalls vorgenommen. Dafür wird eine Herangehensweise verwendet, die häufig beim Vergleich unterschiedlicher Testmethoden vorzufinden ist. Das dafür entwickelte Verfahren untersucht die Entwicklung des Konfidenzintervalls der Bewertungen von einem Stimuli in Abhängigkeit von der Anzahl der Bewertungen. Dafür wird wiederholt eine ausgewählte Menge k aller Probanden n gewählt und das Konfidenzintervall auf Basis dieser Auswahl k berechnet. Da hierbei im Idealfall alle $\binom{n}{k}$ Kombinationen berechnet werden müssten, wird aus Komplexitätsgründen die Berechnung mit 300 zufällig gewählten repräsentativen Kombinationen durchgeführt. Das Resultat dieser Berechnungen kann in Abb. 34 betrachtet werden.

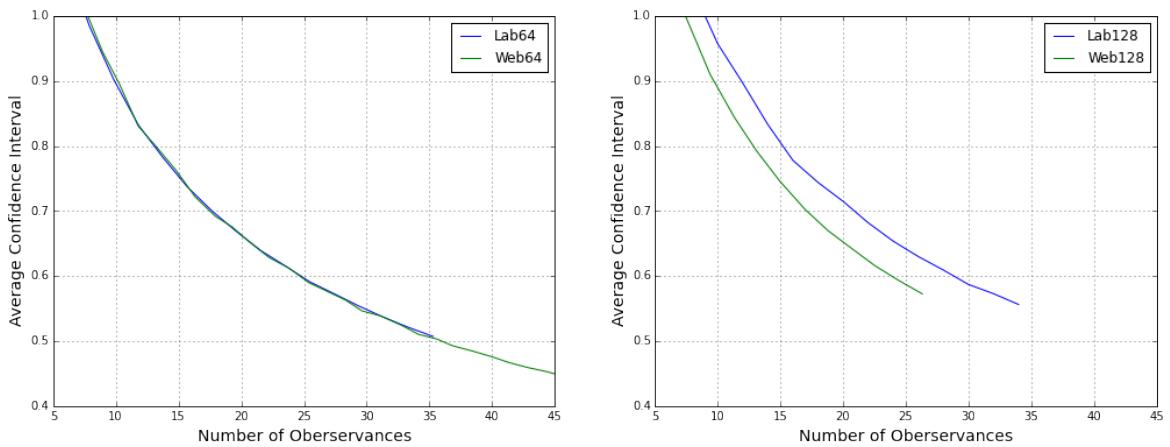


Abbildung 34: Entwicklung des Konfidenzintervalls in Abhängigkeit von Anzahl der Bewertungen für kleine Stimuli (links) und größere Stimuli (rechts) [G22]

Während sich die Experimente mit kleinen Stimuli und das Webexperiment mit großen Stimuli hierbei weitestgehend gleich verhalten, ist beim Laborexperiment mit großen Stimuli ein deutlicher Anstieg zu sehen. Das bedeutet, dass bei diesem Laborexperiment eine geringere Einigkeit bei der Bewertung der Stimuli vorliegt, als bei allen anderen Assessments. Die ursprüngliche Erwartung daran war ein besseres Konfidenzintervall durch den Labortest zu erreichen, da Bildkontext, Störungen und Details besser zu erkennen sind.

Um diesen Zusammenhang näher zu untersuchen wurde dafür Abb. 35 erstellt. Bei diesem Streudiagramm wird der *Mean Opinion Score* mit dem mittleren Konfidenzintervall in einen Kontext gesetzt. Ein deutliches, bogenartiges Abfallen des Konfidenzintervalls ist an den Enden der Bewertungsskala zu entdecken. Je weiter außen die durchschnittliche Bewertung eines Stimuli auf der Skala liegt, desto einheitlicher fällt auch die Bewertungen dafür aus. Eine große Uneinigkeit zeigt sich unterdessen im mittleren Feld der Skala. Diese scheinbar komplizierteren Testfälle bekommen im Durchschnitt eine weniger eindeutige Bewertung. Dies könnte bspw. eine mögliche Erklärung für den größeren Konfidenzintervall der großen Stimuli sein. Denn die zusätzlichen Informationen und Details, die in den größeren Bildern enthalten sind, machen unter Umständen die Aufgabenstellung schwieriger für den Probanden, was sich in einem größeren Konfidenzintervall widerspiegelt. In Abb. 35 ist außerdem auch der erhöhte Konfidenzintervall für größere Stimuli zu erkennen und die Tatsache, dass sowohl große als auch kleine Stimuli eine ähnliche durchschnittliche Bewertung erhalten.

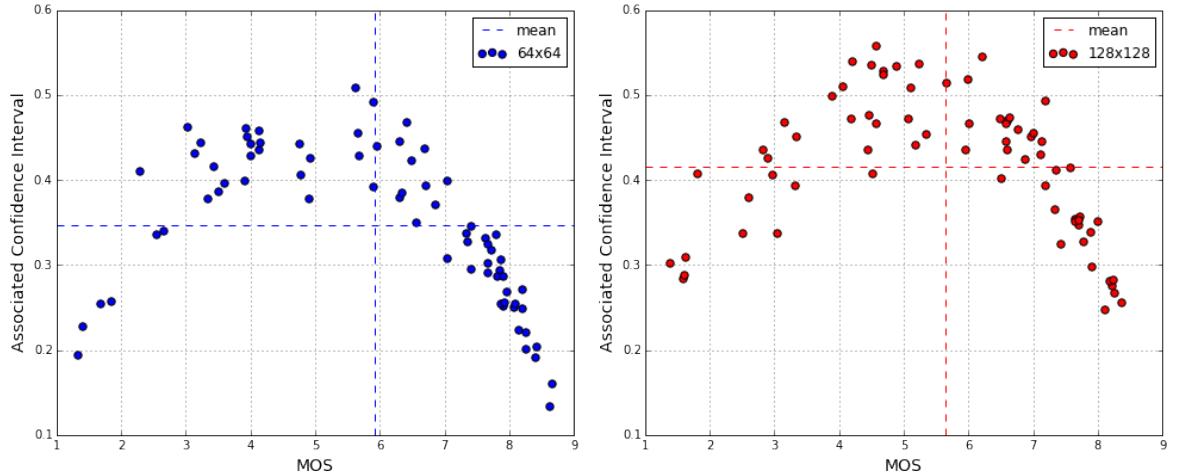


Abbildung 35: Streudiagramme zur Betrachtung der Korrelation zwischen MOS und den beiden Stimuligrößen [G23]

Vergleich der unterschiedlichen Stimuligrößen

Für den weiteren Vergleich der Bewertungen von Stimuli unterschiedlicher Größe werden die gemeinsamen Textur-Stimuli aus den beiden Laborexperimenten miteinander verglichen. Die dafür erstellte Abb. 36 fasst dafür die Durchschnittsbewertungen der unterschiedlichen Qualitätsstufen in einer Grafik zusammen. Die tendenziell schlechtere Bewertung für große Stimuli ist ein Indiz für eine bessere Fehlerwahrnehmung der Probanden. Allerdings zeigen einige Sequenzen, wie bspw. *oatmeal1-a_lumAdjusted*, einen deutlichen Bewertungsunterschied für einige Qualitätsparameter und andere eine sehr identische Bewertung beider Bildgrößen.

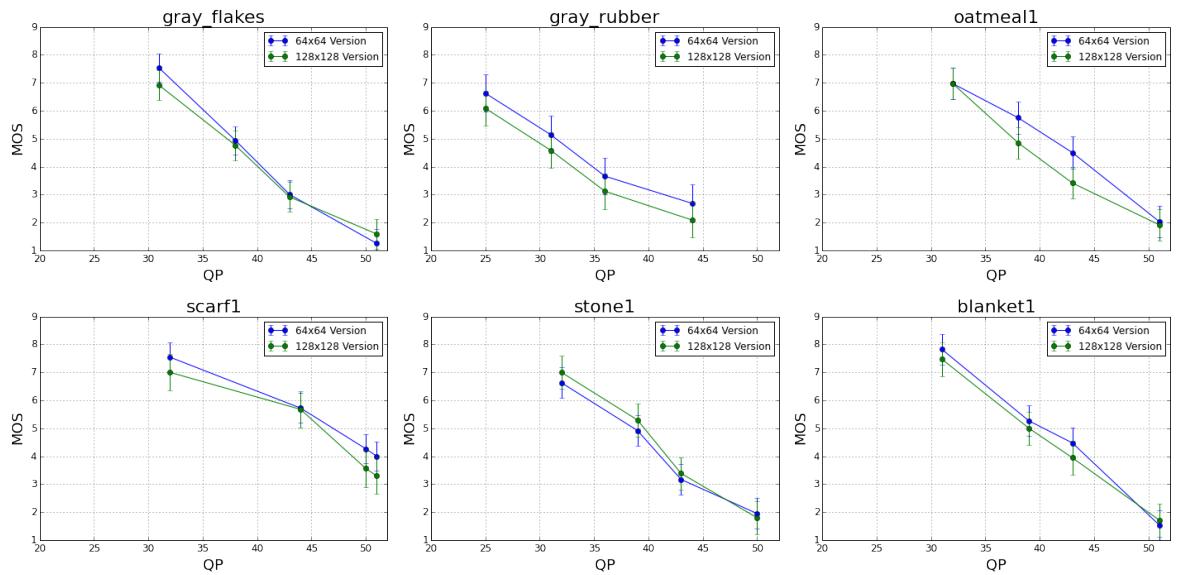


Abbildung 36: Bewertungsentwicklung für unterschiedliche Bildgrößen [G24]

Zusammenfassung

Zusammenfassend können mit dieser Untersuchung die drei Fragestellungen beantwortet werden. Das in den Grundlagen erarbeitet Problem der fehlenden Korrelation zwischen subjektiven und objektiven Bewertungsmethoden konnte bestätigt werden und zeigt eine Übereinstimmung mit den Werten aus der Literatur. Diese Übereinstimmung soll an dieser Stelle auch zur Validierung des entwickelten Systems dienen.

Darüber hinaus konnten zudem sinnvolle und vergleichbare Resultate in den crowdbasierten Assessments über das Internet erzielt werden. Zwar zeigen diese Bewertungen eine systematische Abweichung von den Laborresultaten, haben aber einen linearen Zusammenhang zu diesen Referenzergebnissen. Unter Berücksichtigung der Vorteile und mit einer umfangreichen Analyse der Bewertungen und Probanden, können *crowdbasierte Assessment* so eine sehr gute Alternative zur traditionellen Vorgehensweise sein. Ein Plausibilitätsprüfung mit äquivalenten Assessments in Laborumgebung kann bei Ihnen für die nötige Glaubwürdigkeit sorgen.

Die letzte Fragestellung über die unterschiedlichen Größen liefert an dieser Stelle leider keine klaren Ergebnisse. Die größeren Stimuli zeigen in einigen Fällen eine Abweichung von der ursprünglichen Bewertung der kleineren Stimuli. Der Abstand zwischen diesen Bewertungen ist allerdings sehr klein und die Mittelwerte liegt in vielen Fällen noch im Konfidenzintervall des jeweiligen Vergleichspunktes. Ein interessanter Fakt ist darüber hinaus, dass im Labor bei den größeren Stimuli auch ein Anstieg der Bewertungsunsicherheit zu sehen ist. Unter Umständen deutet dies auf eine bessere Fehlerwahrnehmung hin, die sich durch eine weitere Zunahme der Bildgröße noch verstärken könnte.

7 Fazit und Ausblick

In dieser Arbeit konnte eine funktionstüchtige Webapplikation zur Durchführung von ITU-konformen und selbst erstellten Bild- und Videoqualitätstests umgesetzt werden. Für eine abschließende Zusammenfassung, soll an dieser Stelle der Projektvergleich in Abb. 37 aus [TH14] herangezogen werden. Die implementierte Anwendung ermöglicht, wie die Meisten der in Abb. 37 aufgelisteten Projekte, die Nutzung von Bild- und Videodaten und beherrscht die grundlegenden Verfahren aus [IT12], [IT08] und [IT07a]. Einige der Projekte bieten außerdem noch die Möglichkeit Audioqualitätstests durchzuführen, die in dieser Arbeit keine Rolle gespielt haben. Im Gegensatz dazu können mit der Anwendung individuelle Nutzerumfragen durchgeführt und mit wenig Aufwand auch eigene Testverfahren erstellt werden, was bei vielen der aufgelisteten Projekte nicht oder nur beschränkt möglich ist. Auf die Anordnung der Präsentationsreihenfolge der Stimuli hat der Anwender allerdings keinen Einfluss, dies könnte aber durch ein zusätzliches Attribut in der Datenbankrelation schnell behoben werden. Zur Datenanalyse bei den Projekten in Abb. 37 wird keine Aussage getroffen, da diese vermutlich auch durch externe Programme vorgenommen wird. Dieses Vorgehen hat sich zumindest bei dieser Umsetzung als sinnvoll erwiesen, da die Zielsetzungen von unterschiedlichen Experimenten sehr voneinander abweicht und dafür zugeschnittene Analyseverfahren benötigt werden. Mit Hilfe der IPython Notebooks kann dabei eine individuelle Auswertung nach eigenen Vorstellungen realisiert werden und trotzdem auf einen umfangreichen Satz an bereits implementierten Funktionen zurückgegriffen werden. Des weiteren kann keins der aufgeführten Projekte eine Glaubwürdigkeitsprüfung nach ITU Richtlinien vorweisen, jedoch werden in einigen Fällen andere Screening-Methode umgesetzt. Ein letzter wichtiger Punkt beim Vergleich

der unterschiedlichen Projekte ist die Speicherung der Daten. Bei der umgesetzten Applikation erfolgt dies in einer eigenständigen Datenbank, was eine effiziente und konsistente Verwaltung der anfallenden Datenmengen ermöglicht und somit eine bessere Lösung darstellt als die Verwendung von Textdateien.

Framework Feature \	Euphoria [12]	CrowdMOS [6]	QualityCrowd2 [19]	WESP [20]	BeagleJS [22]	<i>in-momento</i> [23]
Media types	Image, video & audio	Image, audio	Image, video & audio	Image, video, audio, sensory effects	Audio	Image, video
Methodology	PC (binary scale)	ACR, DCR, MUSHRA	ACR, flexible: single & double stimulus; discrete & continuous scales	All (flexible), e.g., ACR, ACR-HR, DSCQE, Double stimulus for sensory effects	ABX, MUSHRA	ACR
Questionnaires	None	Embedded in evaluation	Separated tasks	Embedded in evaluation	None	None
Tasks design	Fixed template	Custom template All tasks have the same template	Custom template Tasks configured in script file	All tasks have the same template	Fixed template	Fixed template
Tasks order	Random All pairs	Random Full set or subset of all stimuli	Fixed	Flexible	Fixed	Random Based on actual number of ratings
Screening	Transitivity index	95% CIs	None	None	None	Reliability profile
Data storage	Text files	Text files	Text files CSV format	Database	Text files	Database
Open source	No ¹²	Yes ¹³	Yes ¹⁴	Yes ¹⁵	Yes ¹⁶	Yes ¹⁷
Programming language	N/A	Ruby	PHP + own script language	Javascript + PHP	Javascript + PHP	PHP

Abbildung 37: Vergleich von Frameworks für Bild-, Video- und Audioqualitätstests [TH14]

In der Zukunft soll die Applikation noch durch einige Arbeitsschritte verbessert werden. Dazu zählt bspw. die Erweiterung und genauere Erprobung der Videounterstützung mit neuen Verfahren und einer modifizierten Vorladetechnik. Eine weitere Aufgabe stellt auch die Verbesserung der Screening-Methode dar, da besonders das zweite Ausschlusskriterium in der Untersuchung klare Schwächen gezeigt hat. Darüber hinaus sollte die zuvor erwähnte Möglichkeit für eine festdefinierte Präsentationsreihenfolge implementiert werden. Für eine bessere Bedienbarkeit könnte außerdem eine Grafische Oberfläche zur Konfiguration erstellt werden, wenn dies benötigt wird. Viele andere Verbesserungen lassen sich außerdem im Bezug auf Sicherheit und Skalierbarkeit finden, was bisher beim Projekt keine Rolle gespielt hat. Zunächst stellt aber auch die weitere Untersuchung der gewonnenen Daten eine wichtige Aufgabe dar.

Letztlich konnte die implementierte Anwendung jedoch bereits seine Funktionstüchtigkeit in einem ersten Experiment mit mehr als 10.000 Bewertungen und einer Vielzahl an Probanden erfolgreich unter Beweis stellen. So kann daher abschließend von einer geeigneten Umsetzung der Aufgabenstellung gesprochen werden, die auch neue Ansätze und Ideen beinhaltet und somit hoffentlich die Mitarbeiter am Fraunhofer Heinrich Hertz Institut bei zukünftigen Fragestellungen gut unterstützen wird.

Literatur

- [DY09] Richard Dosselmann and Xue Dong Yang. A comprehensive assessment of the structural similarity index. *Signal, Image and Video Processing*, 5:81–91, November 2009.
- [IT07a] ITU-T. Methodology for the subjective assessment of video quality in multi-media applications BT.1788. 2007.
- [IT07b] ITU-T. Specifications and alignment procedures for setting of brightness and contrast of displays BT.814. 2, 2007.
- [IT08] ITU-T. Subjective video quality assessment methods for multimedia applications P.910 (04/2008). 2008.
- [IT12] ITU-T. Methodology for the subjective assessment of the quality of television pictures BT.500-13. 13, 2012.
- [LJ11] Weisi Lin and C.-C. Jay Kuo. Perceptual visual quality metrics: A survey. *Journal of Visual Communication and Image Representation*, 22(4):297–312, may 2011.
- [Net15] Mozilla Developer Network. Web API Interfaces. <https://developer.mozilla.org/de/docs/Web/API/Window/devicePixelRatio>, 2015. Accessed March 28, 2015.
- [Sen15] SensioLabs. The Symfony Book. <http://symfony.com/doc/current/book/index.html>, 2015. Accessed March 10, 2015.
- [TH14] Pavel Korshunov Tobias Hoßfeld, Matthias Hirth. Survey of Web-based Crowdsourcing Frameworks for Subjective Quality Assessment. *Multimedia Signal Processing (MMSP 2014)*, 2014.
- [VA11] C. Sasi Varnan and Dr.D.S.Rao A.Jagan, Jaspreet Kaur, Divya Jyoti. Image Quality Assessment Techniques pn Spatial Domain. *IJCST, September 2011*, 2:177–184, 2011.
- [VQE10] VQEP. Report on the Validation of Video Quality Models for High Definition Video Content. 2010.
- [VS14] Gary J. Sullivan et al. Vivienne Sze, Madhukar Budagavi. *High Efficiency Video Coding (HEVC)*. Springer, 2014.
- [WaBSS04] Z. Wang, a.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004.
- [WB09] Zhou Wang and Alan C Bovik. Mean Squared Error: Love It or Leave It? *IEEE Signal Processing Magazine*, January 2009, pages 98–117, 2009.
- [Win13] Stefan Winkler. *Digital Video Quality: Vision Models and Metrics*. Wiley, 2013.

- [WM08] Stefan Winkler and Praveen Mohandas. The evolution of video quality measurement: From PSNR to hybrid metrics. *IEEE Transactions on Broadcasting*, 54:660–668, 2008.

Grafiken

- [G1] Jeff Dahl. Snellen Chart. http://upload.wikimedia.org/wikipedia/commons/9/9f/Snellen_chart.svg. [Online; accessed 16-12-2014].
- [G2] Ted M. Montgomery. pseudoisochromatic plates 3 and 11. http://www.tedmontgomery.com/the_eye/colortst/colortst.html. [Online; accessed 16-12-2014].
- [G3], [G4] sind Beispiele aus den erstellten Python Skripts.
- [G5], [G6] Selbst erstellte Grafik.
- [G7] Modifizierte Grafik von <http://goo.gl/9RfrV5>. [Online; accessed 26-03-2015].
- [G8] Ein Screenshot aus dem Projektverzeichnis in Eclipse.
- [G9] Eine erweiterte Variante der Grafik auf Seite 11 in [Sen15].
- [G10] Ein Screenshot aus dem Reverse Engineering Tool der MySQL Workbench.
- [G11] Ein Screenshot aus dem Python Notebook für die Datenanalyse.
- [G12] Ein Foto aus dem Labor.
- [G13] Eine erstellte Grafik aus dem Kalibrierungsprotokoll der Spyder 3 Software.
- [G14] Eine selbst erstellte Grafik.
- [G15] bis [G18] Testsequenzen für das Projekt vom Fraunhofer HHI.
- [G19] bis [G24] Grafiken aus den Python Notebook für die Datenanalyse.

Hinweis: Alle selbst erstellten und viele weitere Grafiken der Untersuchung sind auch auf dem beigelegten Datenträger zu finden.

Abkürzungsverzeichnis

ACR	Absolut Category Rating
API	application programming interface
CFS	Contrast Sensitivity Function
CIE	Commission Internationale de l'Éclairage
CRT	Cathode Ray Tube
CSS	Cascading Style Sheets
DBMS	Database Management System
DCR	Degradation Category Rating
DCT	Discrete Cosine Transformation
DMOS	Differential Mean Opinion Score
DSCQS	Double Stimulus Continuous Quality Scale
DSIS	Double Stimulus Impairment Scale
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HVS	Human Visual System
HVS	Human Visual System
IFC	Information Fidelity Criterion
IQM	Image Quality Metric
ITU	International Telecommunication Union
JSON	JavaScript Object Notation
LCD	Liquid Crystal Display
MAE	Mean Absolute Error
MOS	Mean Opinion Score
MOS	Mean Opinion Score
MSE	Mean Squared Error
MSVD	Multi-resolution Singular Value Decomposition
MVC	Model View Controller
PHP	Hypertext Preprocessor
PLUGE	Picture Line-Up Generation Equipment
PSNR	Peak Signal-to-noise Ratio
PVD	Preferred Viewing Distance
PVQM	Perceptual Visual Quality Metrics
SAMVIQ	Subjective Assessment of Multimedia VIdeo Quality
SNR	Signal Noise Ratio
SQL	Structured Query Language
SSIM	Structural SIMilarity index
SSMR	Single Stimulus with Multiple Repetition
VIF	Visual Information Fidelity
VSNR	Visual Signal-to-Noise Ratio

Dateiname	Größe	QP	PSNR
blanket1.png	64	31, 39, 43, 51	28.17, 28.02, 27.95, 27.93
gray_flakes001-120dpi.png	64	31, 38, 43, 51	27.72, 27.64, 27.66, 27.69
gray_rubber001-300dpi.png	64	25, 31, 36, 44	26.80, 26.86, 26.85, 26.85
oatmeal1.png	64	32, 38, 43, 51	28.27, 28.19, 28.06, 28.03
scarf1.png	64	32, 44, 50, 51	27.79, 27.83, 27.89, 27.86
stone1.png	64	32, 39, 43, 50	27.54, 27.60, 27.57, 27.77
EbuA_lawn_fr0_6.png	64	21, 26, 34, 40	42.68, 38.27, 33.45, 31.92
SesA_fr2_401.png	64	25, 29, 35, 39	45.71, 43.02, 38.87, 37.02
SesB_fr1_358.png	64	21, 27, 30, 36	42.65, 37.62, 35.44, 32.83
SesC_fr1_3.png	64	25, 28, 32, 38	40.43, 38.04, 35.40, 32.87
SesD_fr0_30.png	64	21, 25, 31, 36	43.77, 40.58, 36.01, 33.44
SesE_fr0_357.png	64	20, 23, 31, 40	45.75, 43.99, 40.12, 37.24
SesE_fr1_325.png	64	25, 29, 35, 40	45.78, 44.43, 40.51, 38.66
SesE_fr1_359.png	64	24, 29, 32, 38	44.25, 40.50, 38.46, 35.00
SkyA_fr0_304.png	64	23, 27, 34, 40	41.29, 37.86, 33.46, 31.79
SkyA_fr0_343.png	64	25, 30, 35, 40	41.05, 38.07, 35.74, 33.60
SkyA_fr1_95.png	64	20, 25, 28, 38	44.93, 41.12, 38.77, 33.79
SkyA_fr2_289.png	64	22, 27, 32, 38	45.64, 43.30, 40.62, 38.10
blanket1.png	128	31, 39, 43, 51	28.12, 28.04, 28.03, 27.96
gray_flakes001-120dpi.png	128	31, 38, 43, 51	27.82, 27.80, 27.79, 27.82
gray_rubber001-300dpi.png	128	25, 31, 36, 44	26.78, 26.91, 26.96, 26.86
oatmeal1.png	128	32, 38, 43, 51	28.17, 28.12, 28.03, 27.96
scarf1.png	128	32, 44, 50, 51	27.84, 27.88, 27.90, 27.89
stone1.png	128	32, 39, 43, 50	27.53, 27.57, 27.63, 27.68
EbuA_fr1_57.png	128	22, 27, 32, 38	45.95, 43.19, 40.14, 37.69
EbuA_lawn_fr0_65.png	128	23, 26, 31, 36	41.31, 39.17, 36.46, 34.71
SesC_fr0_119.png	128	23, 27, 35, 39	42.59, 39.70, 34.83, 33.32
SesD_fr1_113.png	128	22, 28, 34, 40	42.70, 37.79, 34.04, 32.13
SesE_fr1_0.png	128	23, 27, 35, 40	43.27, 40.50, 36.40, 34.01
SesE_fr1_118.png	128	23, 27, 31, 39	45.05, 42.86, 40.68, 36.87
SesE_fr1_44.png	128	23, 30, 34, 38	45.65, 40.93, 38.49, 36.47
SesE_fr1_91.png	128	25, 29, 34, 39	40.59, 37.56, 34.72, 32.72
SkyA_fr0_25.png	128	23, 27, 33, 39	41.11, 37.91, 33.97, 31.93
SkyA_fr1_25.png	128	22, 26, 30, 37	42.97, 39.85, 37.06, 33.89
SkyB_fr0_24.png	128	23, 27, 31, 39	41.92, 38.88, 36.15, 32.48
SkyB_fr0_86.png	128	21, 25, 29, 34	45.34, 42.66, 41.04, 39.60

Tabelle 1: Vollständige Liste aller Stimuli aus dem Experiment mit Größe, QP und PSNR

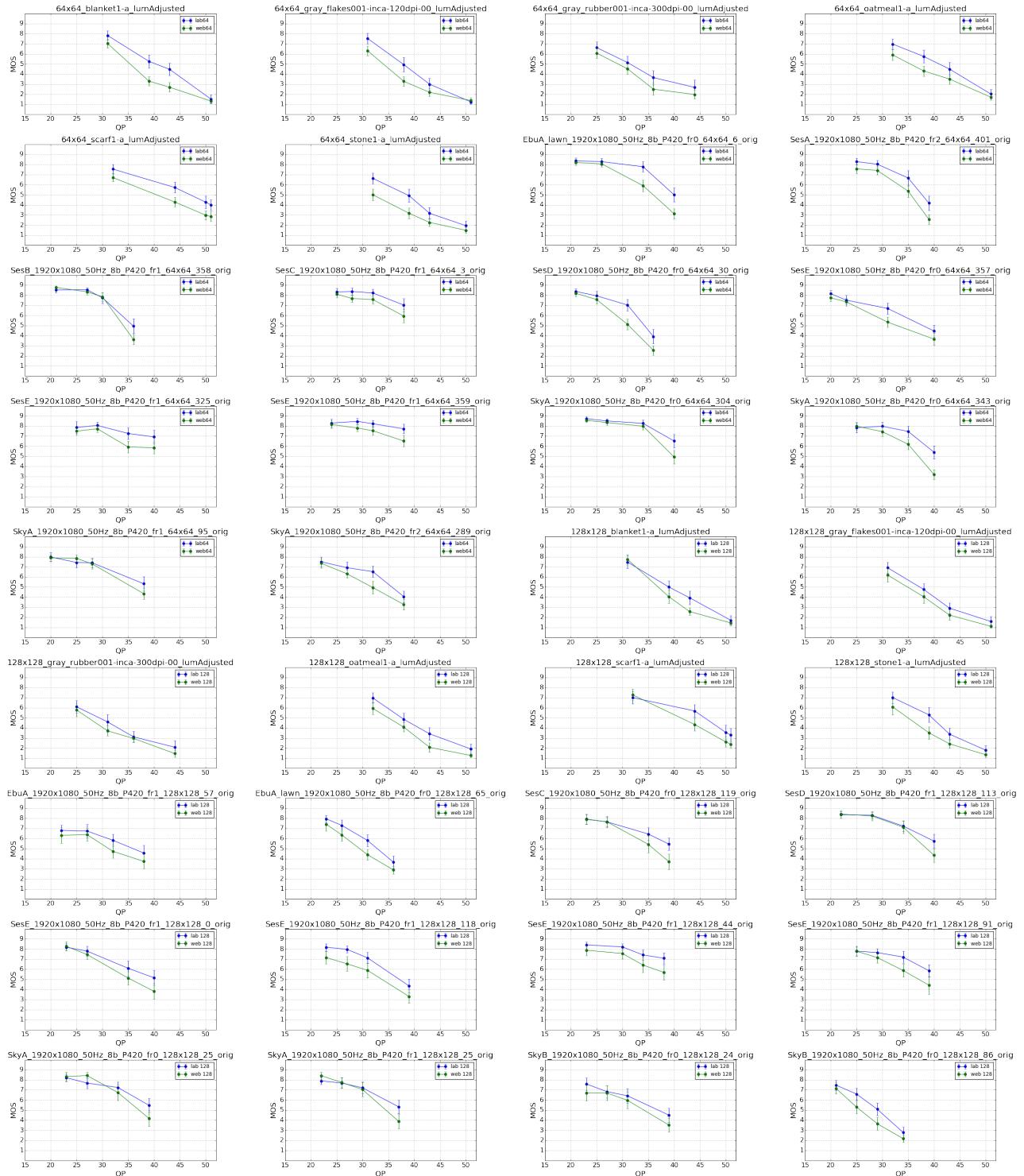


Abbildung A1: Grafiken zur Analyse des Zusammenhangs zw. MOS und QP mit Angabe des Konfidenzintervalls (95%). [G16]