

ISIDOR – Ein auditiver HCI-Prototyp

Josefine Riedel, Thomas Böck, Julian Fischer, Felix Rauchwarter

St. Pölten University of Applied Sciences
E-Mail: dm161537@fhstp.ac.at

Abstract—The work shown engages with the conception and realization of the human-computer interaction (HCI) sound installation ‘Isidor’ which gives the user the possibility to conduct a virtual string quartet. The control system appears via a leap motion camera (motion tracking) and a clearly arranged user interface (GUI) implemented on a touch screen. Emanating from the GUI parameters as volume, soli and mutes, room impressions, tonal pitches (Hz), moods (emotional), the changing of instrumental player positions and the punctual launch from a certain position are possible. The leap motion camera acts like a translator from the conductor movements of the user’s digital workable control data. Therefor working in real time is essential for tempo changes. The Max/MSP-object Bachscore is accountable for the score implementation. MIDI data is transferred to VST-instruments which eventually are responsible for the acoustic sound. Max/MSP patches connect the different elements: Leap motion tracking, DMX light control, graphics and Bachscore to an interactive sound installation.

Zusammenfassung—Die vorliegende Arbeit beschäftigt sich mit der Konzeption und Umsetzung der Human-Computer Interaction (HCI)-Klanginstallation “Isidor”. Sie ermöglicht es Nutzern, ein virtuelles Streichquartett zu steuern. Der Benutzer interagiert mittels Motion Tracking der Hände durch eine Leap Motion Kamera sowie einer übersichtlich gestalteten Nutzeroberfläche auf einem Touchbildschirm. Ausgehend von dieser lassen sich Parameter wie Lautstärke, Besetzung, Raumeindrücke, tonale Stimmung (in Hz), Gefühlszustände (“Moods”), Positionsänderungen der Instrumentalisten, Artikulation jedes Spielers sowie das punktuelle Abspielen an einer gewünschten Position umsetzen. Die Leap Motion übersetzt Dirigatsgesten des Nutzers in Tempoänderungen, die in Echtzeit generiert werden. Mittels der Max/MSP-Library “Bach” wird die Notendarstellung auf der grafischen Benutzeroberfläche angezeigt. Die Klangerzeugung geschieht durch mehrere VST-Instrumente. Max/MSP Patches vernetzen die verschiedenen Elemente (Leap Motion Tracking, DMX Lichtsteuerung, Grafik und Bachscore) miteinander zu einem interaktiven Klangerlebnis.

I. EINLEITUNG

Unter “Sonic Gestures” versteht man Bewegungen oder Mimiken des menschlichen Körpers, die Auswirkungen auf das Klangbild haben oder sogar Klänge erzeugen können. ‘Sonic’ bedeutet ‘(akustischer) Schall’, während ‘Gesture’ auf die menschliche Komponente, die Körper- bzw. Ausdrucksbewegung hinweist [1]. Das grundsätzliche Prinzip von Sonic Gestures wird aktuell in Form von Klanginstallationen in vielen Museen aufgegriffen. Durch das in-Kontakt-treten mit den technischen Installationen steigt das Interesse des Museumsbesuchers und das subjektive Gefühl der Erlebbarkeit der Installation. Die Folge ist eine unterbewusste und doch geradlinige Ansprache zwischen Rezipient und wissenschaftlicher

Forschungsquelle. Es kann abgeleitet werden, dass eine direkte Interaktion die Aufnahme- und Lernfähigkeit sowie das kognitive Gedächtnis fördert [2]. Von wissenschaftlicher Seite wird dieser Ansatz Sound and Music Computing (SMC) genannt. Dieses Versuchsfeld hat die Prämisse, die Gesamtheit der Klang- und Musikkommunikationskette von einem multidisziplinärem Standpunkt aus zu betrachten. Wissenschaftliche, technologische und künstlerische Methoden sollen zum Ziel des Verstehens, Generierens und Modellierens von akustischen Klängen durch computerbasierte Techniken beitragen [3].

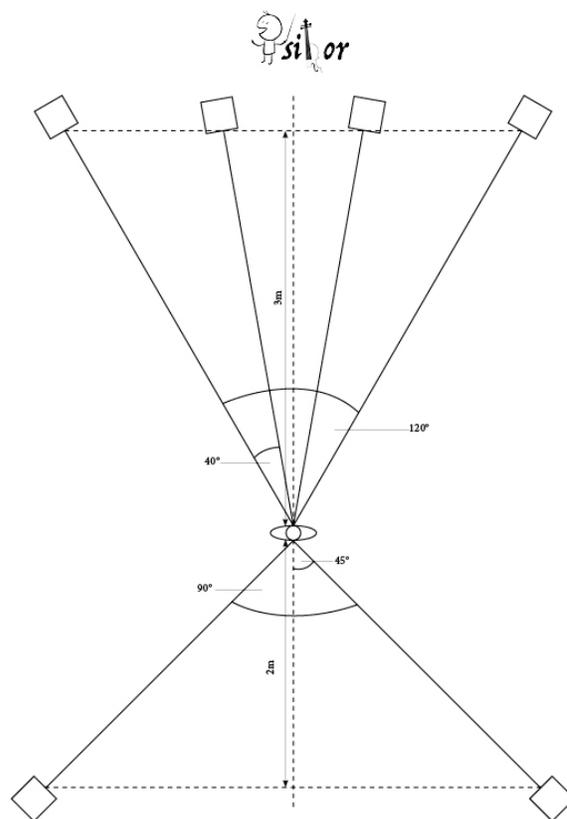


Abbildung 1. Isidor Setup

Mit Isidor soll dieser Ansatz des spielerischen Lernens aufgegriffen und erweitert werden. Die Erklärung des Namens unterstreicht den Grundgedanken der Forschung: ‘Ea-

sy DAW’ (“Digital Audio Workstation”) verbalisiert ‘Isidor’. Das Programm bietet einem breitgefächertem Publikum die Möglichkeit, direkten Einfluss auf die Darbietung eines Streichquartetts auszuüben. Die interaktive Installation setzt sich seitens der Hardware aus einem übersichtlichen User Interface auf einem Touchbildschirm, sechs Lautsprechern (vier Frontkanäle vor dem Rezipienten und zwei Rearkanäle hinter demselben zur Unterstützung der räumlichen Darstellung), einer Leap Motion Tracking Kamera (für die Tempoübersetzung des Dirigats) und via DMX gesteuerten Scheinwerfern (zur Unterstreichung der Tempoinformationen und Atmosphäre) zusammen.

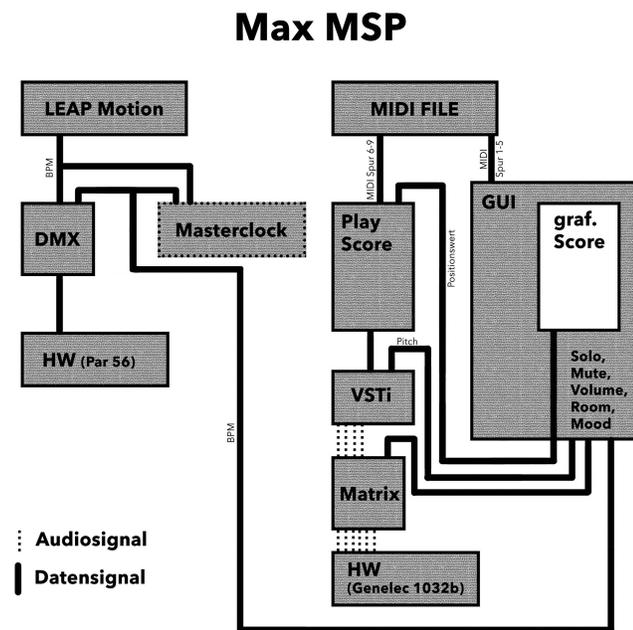


Abbildung 2. Technischer Aufbau Isidor

Softwareseitig wird eine vorprogrammierte MIDI-Datei mittels dem Max/MSP-Objekt Bachscore interpretiert, in Partiturdarstellung angezeigt und an VST Instrumente weitergeleitet, die Klänge generieren. Die Leap Motion Kamera erfasst dirigatsähnliche Handbewegungen des Nutzers, welche in einem separaten Max/MSP-Patch in Beats Per Minute (BPM) umgerechnet werden. An diesen Prozess ist auch die DMX-Steuerung angebunden, die durch farbiges Licht zusätzlich zum Hörerlebnis ein optisches Feedback über die aktuelle Geschwindigkeit des Stückes gibt und, in Folge dessen, die Stimmung intensiviert und das Nutzererlebnis um eine visuelle Ebene erweitert. Den Großteil der Steuerungen übernimmt der Rezipient über das Graphical User Interface (GUI). Ein zentraler Hauptbildschirm ohne Screens ermöglicht schnellen und intuitiven Zugriff mittels Touch-Steuerung auf folgende Funktionen: Transport (Play, Pause), Ausdruck bzw. Lautstärke jedes Instruments (gekoppelt an Note-On Expression), Mute-Funktion für jeden Spieler, Raumeindrücke (Studio/Dry, Small Hall, Big Hall), tonale Stimmung (in Hz),

sogenannte “Moods” (bspw. wütend, verschlafen, vorsichtig), Positionsänderungen der Instrumentalisten sowie das punktuelle Abspielen ab einer gewünschten Position durch “Tap” mit dem Finger in die Partiturdarstellung.

II. STATE OF THE ART

Bereits 2001 wurde die Klanginstallation “Personal Orchestra” in Wien aufgestellt. Sie ist im ‘Haus der Musik’ stationiert, einem Zentrum der musikalischen Kulturbewahrung und -vermittlung. Die Herren Borchers, Samminger und Mühlhäuser behaupten mit ‘Personal Orchestra’ die erste Klanginstallation geschaffen zu haben, die es dem Nutzer ermöglicht, eine Audio-/Videoaufnahme in Echtzeit dirigieren zu können. Bei dem verwendeten Material handelt es sich nicht, wie bei Isidor, um verarbeitete MIDI-Daten, sondern um eine “echte” Tonaufnahme der Wiener Philharmoniker. Die Beeinflussung von Tempo, Lautstärke und Instrumentation ist durch Motion Tracking einer Infrarotkamera ohne hörbare Artefakte, die durch Time Stretching entstehen können, möglich. Hierbei sei jedoch angemerkt, dass die Tempovariation nur in einem eingeschränkten Rahmen umsetzbar ist. Wie bereits erwähnt, wurden alle Stücke aufgenommen und mittels Timestretching offline in verschiedenen Tempi gerendert. Bei einer Tempoänderung durch den Nutzer werden die passenden Audiodateien abgerufen und überblendet. Überschreitet ein Nutzer die Grenzen des geringsten oder höchsten vorgeordneten Tempos, hält die Wiedergabe an und der virtuelle Konzertmeister beschwert sich humoristisch über die Anstrengungen des Dirigenten. Ein weiterer Unterschied zu Isidor besteht darin, dass auch Videodateien abgespielt werden. Diese sind, wie das Audiomaterial, vorher aufgezeichnet und werden in Echtzeit überblendet [4].

Inspiration für die vorliegende Arbeit wurde weiters in der Klanginstallation “Effektorium” gefunden. Die seit 2014 im Leipziger Mendelssohn-Bartholdy Museum beheimatete Installation erlaubt es Nutzern ebenfalls, in die Rolle eines Dirigenten zu schlüpfen und ein virtuelles Orchester anzuführen. Säulenförmige Lautsprecher mit Display stehen stellvertretend für ausgewählte Musikergruppen im Raum. Die Leap Motion Kamera erfasst die Dirigatsgesten, deren ausgewertete Informationen mittels eines Max/MSP-Patches in Verbindung zu Ableton Live stehen. Die Audio-Workstation beheimatet Audiodateien, die in Echtzeit durch Timestretching auf das vorgegebene Tempo gedehnt bzw. gestaucht werden [5], [6].

Auch im Alltag geschieht in jüngerer Vergangenheit vermehrt Datengewinnung und -verarbeitung via Kinect oder Leap Motion-Technik. Kieran Flay widmete sich 2015 einer Gegenüberstellung dieser zwei Arbeitsmittel in Bezug auf die Verbindung von Gestikverarbeitung und digitalen Audiosystemen. Flay vergleicht die Technologien vor allem in Bezug auf die Parameter Durchführbarkeit, Nutzen und Funktionalität. Als weitere Option für gestikgesteuerte Systeme wird auf Smart TVs (Lautstärkesteuerung, Schnittstellenwechsel, Menü- bzw. Browsernavigation) oder Remidi Gloves (ein Handschuh mit Bluetooth-MIDI-Steuerung) verwiesen [7].

III. METHODIK

Ziel des Projektes ist es, eine Installation zu gestalten, die dem Benutzer ein subjektives Kontrollgefühl verleiht und verständlich sowie intuitiv handhabbar ist. Um einen Einblick in den Stand der Forschung und ähnliche Projekte zu erhalten, wurde eine State Of The Art-Recherche durchgeführt. Die Ergebnisse gaben einen Rahmen vor, in dem sich das Projekt bewegen würde. Anschließend wählten die Autoren Technologien und Interfaces aus, welche die Umsetzung möglich machen. Aus dem weiteren Arbeitsfortschritt entstand ein Prototyp, der im Rahmen einer Ausstellung aufgebaut und von Besuchern ausprobiert werden konnte. Das verbale Feedback der Probanden wurde im Nachhinein eingearbeitet und das Installationsdesign dadurch optimiert.

IV. SIGNALFLUSS

Das vorliegende Projekt wurde zur Gänze in Max/MSP entwickelt. Herausgebracht von Cycling74 ist MAX eine grafische Entwicklungsoberfläche, mit der vornehmlich akustische und audiogenerative Projekte umgesetzt werden können. Zur grafischen Notendarstellung wurde die frei zugängliche, externe Max/MSP-Library "Bach" einbezogen¹. BachScore ist ein Freeware Programm, dass von den Komponisten Andrea Agostini und Daniele Ghisi entwickelt wurde. Es handelt sich dabei um ein umfangreiches, traditionelles Notationssystem für Max/MSP. Da der Hauptfokus auf der klassischen Notation, Analyse und Komposition innerhalb der Programmierumgebung liegt, ist eine Anwendung als grafische Partitur nach Aussage der Entwickler eher unüblich und entsprechend (noch) nicht im vollem Maße implementiert.

Der strukturelle Aufbau (siehe Abb. 2) lässt sich am einfachsten anhand der vorprogrammierten MIDI-Datei verfolgen. Die ersten fünf Spuren des Files werden per Bach.Score grafisch im GUI dargestellt, während die Spuren sechs bis neun akustisch wiedergegeben werden. Jegliche Interaktion mit der graphisch angezeigten Partitur sendet wiederum Informationen an den wiedergebenden Score. Um akustische Informationen ausgeben zu können, werden die MIDI-Steuerdaten an einen Sampler (in diesem Falle Native Instruments' KONTAKT-Factory Library) mit den vier Software Instrumenten (VSTi: Standard Soloinstrumente Violine, Viola und Cello) weitergegeben, wodurch vier Audiosignale generiert werden.

Diese VSTi können eine gemeinsame Änderung der akustischen Stimmung (Pitch: Standard = Kammerton a bei 440Hz) aus einem Wahlmen auf der graphischen Benutzeroberfläche heraus erfahren. Die vier Audiostreams durchlaufen eine Folge von Matrizen, um sie auf die sechs vorhandenen, physikalischen Ausgänge zu routen. Diese Matrizen können wiederum vom Nutzer durch die GUI beeinflusst werden (Mute-Funktion, Lautstärke jeder Stimme, Verräumlichung, etc.). Zum besseren Verständnis ist die dynamische Tempogestaltung separat zu betrachten. Ausgangspunkt hierfür ist das Motion Tracking der Leap-Motion-Kamera. Ihre ausgegebenen Werte

werden auf BPM-Werte umgerechnet und dann auf einen Multiplikator für die Masterclock skaliert (genaue Besprechung an späterer Stelle).

Der BPM-Wert wird auch an die DMX-Steuerung weitergegeben. Dadurch wird die Farbwahl der Scheinwerfer der Installation bestimmt. Neben der Temposteuerung durch die Leap Motion Kamera können auch direkt über die GUI die sogenannten "Moods", voreingestellte Kombinationen aus Tempo, Lautstärke und Artikulation, ausgewählt werden.

V. HARD- UND SOFTWARESPEZIFIKATIONEN ZUR UMSETZUNG

A. Audiosignalfluss

1) *Direktschall*: Die vier eigenständigen Inputs (V1, V2, Va, Vc) werden zuallererst an alle vier Frontkanäle gesendet. Dort gelangen sie in eine Matrix, die je nach Eingangswert (1-4) genau ein Signal ausgibt. Diese Eingangswerte sind voreingestellt und können dann in der GUI über ein abfragendes Array getriggert werden (genaue Besprechung folgt), wonach jeweils nur ein Wert an genau einer Position vorkommen kann. Die Stimmen werden also in ihrer Position direkt getauscht.

2) *Diffusschall*: Für jeden Frontkanal existiert eine eigene AUX-Schiene, die in ein Reverb-Plugin speist. Das Ausgangssignal jedes Frontkanals wird dort hinein gesendet und die Signale summiert, ehe sie verhallt werden. Dabei werden die Eingänge nach Position gewichtet (Bsp.: Linker Kanal im linken ReverbMix deutlich stärker vertreten als der Rechte Kanal; die Abstufung erfolgt in jeweils 3dB Schritten). Das generierte Summensignal wird abgegriffen und durch das in Max/MSP integrierte algorithmische Hall-PlugIn 'Gigaverb' für jeden Kanal einzeln prozessiert. Das Raumsignal wird abschließend mit dem trockenen Direktschall-Signal summiert (Diffusschall -6dB zu Direktschall) und an den physikalischen Output weitergeleitet.

In den rückwertigen Kanälen ("Rear"), die zur besseren Simulation eines räumlichen Hörerlebnisses benötigt werden, kommen die Ausgangssignale der Vorderkanäle, also die ursprünglichen Inputs, an. Diese werden miteinander summiert, wobei die beiden diagonal gegenüberliegenden Kanäle je um 3dB abgeschwächt werden. Die Verräumlichung erfolgt ebenso über Gigaverb und entspricht immer exakt demselben Preset der Verräumlichung der Frontkanäle.

Um der Ausbreitungsgeschwindigkeit von Schall im Raum Rechnung zu tragen, werden die Outputsignale der Rearkanäle zeitlich verzögert. Da die Verräumlichung durch Gigaverb algorithmisch geschieht, wird der Zeitfaktor des Delays der Rearkanäle an die jeweilige Nachhallzeit des künstlichen Raumes angepasst. Idealerweise sollte die Delayzeit dabei den Zeitraum, in dem beim Nutzer die reale Erstreflektion der Frontsignale an der rückwärtig raumabschließenden Wand eintrifft, nicht übersteigen.

¹<http://www.bachproject.net>, 04.09.2017

Zu beachten bleibt außerdem, dass trocken aufgenommene VST-Instrumente das Klangerlebnis hinsichtlich Ortbarkeit und Klangqualität bedeutend verbessern.

3) *Mute-Funktion*: Wichtig für das Verständnis der Mute-Schaltung ist das Verhalten der wechselnden Positionierung der Eingangssignale (V1, V2 etc.), denen die statischen Kanäle 1-4 gegenübergestellt sind. Um dem Nutzer das Stummschalten einzelner Instrumente, egal an welcher Position, zu ermöglichen, müssen die Eingangssignale bereits gekappt werden, bevor sie einem Kanal zugewiesen werden. Dazu durchläuft jedes der vier Inputsignale vor dem Kanalrouting eine weitere Matrix. Diese hat jeweils zwei Inputkanäle, die sich zur einfachen Verständlichkeit als Hot und Cold bezeichnen lassen. So liegt an einem Input der Matrix das jeweilige Inputsignal (bspw. V1, V2) an, am Anderen kein Signal ("Stumm").

Nachdem die Verräumlichung aller AUX-Schienen durch die Ausgangssignale der Frontkanäle, also erst zu einem späteren Zeitpunkt im Routing, erfolgt, ist also sichergestellt, dass das Stummschalten eines jeweiligen Instrumentes (= Inputsignales) auch im Raumsignal, sowie den Rearkanälen stattfindet.

4) *Tempo*: Prinzipiell können in Bachscore Tempoänderungen implementiert werden. Problem dabei ist, dass diese Änderungen bereits im Vorfeld bzw. vor dem Abspielen der jeweiligen Position definiert werden müssen. Dabei kann das Tempo zu jedem Taktbeginn per Notenwert in BPM ("Viertelnoten in 120 BPM") definiert werden, wobei eine fließende Änderung des Tempos zum nächsten Takt per *accelerando* und *ritardando* möglich ist. Da das Konzept von Isidor jedoch verlangt, Tempoänderungen auch in Echtzeit zu ermöglichen, wird die Masterclock angegriffen, welche die Länge eines Taktes in Samples angibt und somit einen absoluten Zeitwert bietet, der moduliert werden kann.

Dabei muss die aktuelle Geschwindigkeit des Stückes durch die gewünschte (neue, durch das Dirigat bestimmte) BPM-Zahl dividiert werden, um den korrekten Multiplikator für die Masterclock zu erhalten (Bsp: 130 BPM Originaltempo zu 130 BPM anliegendes Tempo = Multiplikator 1).

5) *Pitch*: Das Streichquartett kann mittels einer Master-Kontrolle (d.h. gültig für alle vier Instrumente) in der Stimmung hinsichtlich des Referenztons variiert werden. Zur Auswahl stehen die internationale Standardstimmung auf 440Hz, die historische Stimmung auf 415Hz, die klassische Stimmung auf 430Hz und die Wiener Stimmung auf 466Hz. Die Änderung der Tonhöhe geschieht durch die Pitch Bend-Funktion des VST-Instrumentes. Problematisch dabei ist die Umrechnung von Hertz auf Cent auf linearer Basis, da sich die Tonhöhe logarithmisch verhält. Näherungsweise lässt sich für den relevanten Frequenzbereich um 440Hz ein Verhältnis von rund +/- 4Cent pro einem Hertz Verstimmung annehmen [8].

6) *Darstellung vs. Klang in Midi*: Beim Umgang mit Midinoten ist festzustellen, dass eine strikt notationsgetreue MIDI-Programmierung für die klassische Notendarstellung (Partitur) unumgänglich ist, die akustische Wiedergabe dieser Programmierung allerdings üblicherweise als sehr statisch und unrealistisch wahrgenommen wird (beispielsweise ergeben überlappende Midinoten einen flüssigeren Übergang und damit einen besseren Legato-Eindruck).

Aus diesem Grund wird ein Midifile verwendet, das aus insgesamt neun Einzelspuren besteht. Die ersten vier Spuren entsprechen einer exakten, notationsgetreuen Programmierung für die visuelle Darstellung jeder einzelnen Stimme. Die fünfte Spur ist für die Darstellungsbreite des Partiturfensters notwendig, worauf im Punkt "Darstellung der Partitur" nochmals konkret Bezug genommen wird. Die verbleibenden vier Spuren werden an den Sampler weitergeleitet und akustisch umgesetzt.

Darstellung der Partitur: Eine seitenweise Darstellung mittels *Bach.Score* ist zum aktuellen Zeitpunkt (noch) nicht möglich. Als Workaround werden daher in Isidor vier *Bach.Score*-Instanzen genutzt. Die Instanz, die tatsächlich abgespielt wird und Daten an den Sampler weitergibt, arbeitet versteckt im Hintergrund. Drei weitere Instanzen bilden untereinander positioniert die in der GUI sichtbare Partitur. Jede der drei Instanzen entspricht dabei einem Notensystem mit je 2 Takten. Entsprechend der eben diskutierten Umstände werden nur die ersten fünf Spuren grafisch wiedergegeben.

Prinzipiell folgt die Ansicht in *Bach.Score* immer dem Playmarker, also der aktuellen Abspielposition. Nachdem die grafischen Instanzen allerdings als passiv ("nicht abspielend") verstanden werden können, wird das händische Umblättern möglich.

Bei *Bach.Score*-Instanzen kann die Position des Playmarkers über die Message *inscreenpos X.Y* bestimmt werden, wobei X den Schlag angibt; der Taktbeginn entspricht dem X-Wert 0, während Y die Taktzahl angibt. Da insgesamt sechs Takte gleichzeitig zu sehen sind (zwei pro Instanz), werden bei jeder anliegenden grafischen *Score*-Instanz beim Blättern je 6 Takte zur aktuellen Position dazu addiert. Demnach springt die oberste Instanz von Takt 1 zu 7, von 7 zu 13, ... während die mittlere Instanz von 3 zu 9, von 9 zu 15, ... und das unterste System von Takt 5 zu 11, 11 zu 17, ... springt. Damit nicht über den Start- und Endpunkt hinaus geblättert werden kann, wurden für die Positionswerte per *Clip*-Objekt Grenzen definiert. Sind diese Werte erreicht, kann keine weitere Addition bzw. Subtraktion erfolgen.

Damit dieses Prinzip eines beständigen Sprunges der Ansichtsposition um sechs Takte überhaupt möglich ist, muss sichergestellt werden, dass pro *Score*-Instanz (also Notensystem) immer genau gleich viele Takte dargestellt werden und diese, auch aus grafisch-ästhetischen Aspekten, stets gleich groß sind. Die dargestellte Breite eines Taktes lässt sich in *Bach* per Default nicht einstellen. Dadurch erklärt sich die Notwendigkeit einer "Ansichtsspur" (Spur 5) in der programmierten MIDI-Datei. Diese Spur enthält einen - im Vergleich zum restlichen Notenmaterial - sehr geringen Notenwert, der

konstant wiedergegeben wird. Im vorliegenden Fall entspricht dies einem Notenwert von zweiunddreißigstel Noten. Die benötigte Breite zur graphischen Darstellung dieser zweiunddreißig Einzelnoten, welche kontinuierlich ist, ermöglicht somit das Festlegen einer generellen Taktbreite. Da diese Spur einzig und allein zu diesem Zweck benötigt wird, ist sie im GUI durch ein weißes Viereck überdeckt, um dem User nur die vier hörbaren Instrumente im Notensystem anzuzeigen.

Das Springen zu einer Position durch Touch-Befehl auf den jeweiligen Takt basiert auf demselben Prinzip wie das Blättern. Wiederum werden "Inscreenposition"-Werte ausgegeben. Diese werden an der jeweiligen grafischen Score-Instanz ausgelesen und an die abspielende Instanz weitergegeben, die dadurch direkt zum entsprechenden Takt springt und die Wiedergabe an der gewählten Position startet.

B. Grafik und Design

Aus verschiedenen Entwürfen für die grafische Benutzeroberfläche wurde eine Lösung ausgewählt, die auf drei Spalten basiert: Kontrollen für alle Instrumente, Kontrollen für Einzelinstrumente und Kontrollen für die Partiturdarstellung.



Abbildung 3. Grafisches User Interface von Isidor

In den Kontrollen für Einzelinstrumente findet der Benutzer eine bildliche Darstellung des Instruments, dessen Namen, einen Fader für die Lautstärke des Instruments sowie einen Button um die Stimme stumm zu schalten. Auf der linken Seite der Instrumentenabbildung gibt es außerdem ein Wahlmenü, in dem die Artikulation des Spielers (Legato, Staccato, Pizzicato, Tremolo) ausgesucht werden kann.

Hierbei sei angemerkt, dass der Volumefader nicht den Pegel des Ausgangssignals moduliert, sondern den MIDI-Control-Change Expression des VSTi ansteuert. Dementsprechend wird nicht nur beispielsweise die Lautstärke verringert, sondern auch die musikalische Artikulation angepasst (geringere Expression führt zu geringerem Druck des Bogens auf die Saiten, was in einem weicheren Klangbild resultiert), wodurch ein deutlich realistischeres und natürlicheres Klangbild erzeugt wird.

Von einer zuerst angedachten Lösung, über das Bild des Instruments in einen Subscreen mit erwähnten Kontrollen zu gelangen, wurde abgesehen, um die Kontrollelemente auf einem Blick ersichtlich zu behalten und eine Verschachtelung des Interfaces zu vermeiden. So soll es dem User auch einfacher gemacht werden, die Lautstärkenverhältnisse der Stimmen im Blick zu behalten und eine mögliche Quelle für Verwirrungen (bspw. versteckte Mute-Buttons) ausgeschlossen werden.

Eine weitere Entscheidungsgrundlage für die Anordnung ohne Subscreens war im Lauf der Entwicklung der Installation das Festlegen des verwendeten Touch-Bildschirmes. Hier sei erwähnt, dass viele dieser Entscheidungen bei der Umlegung auf andere Systeme und Kontrolleinheiten (bspw. Smartphone, Tablet) überdacht bzw. angepasst werden könnten.

In den Kontrollen für alle Instrumente (Room, Moods, Pitch) sind die Auswahlmenüs mit Toggle-Buttons ausgeführt. Interessanter ist die Positionsauswahl der Instrumente; hier gibt es eine Matrix, in der man für jede Stimme zwischen Links, Halblinks, Halbrechts und Rechts entscheiden kann. Beim Umsetzen einer Stimme tauscht diese den Platz mit dem Instrument an der jeweils neuen Position. Ursprünglich war angedacht, die Positionen mit 4 Monitoren, die jeweils unter den Front-Lautsprechern stehen, optisch darzustellen. Davon wurde abgesehen und stattdessen eine kleine, bildliche Darstellung jedes Instrumentes über der Positionsauswahl-Matrix eingefügt, die der Sitzordnung folgt. Hierbei war der Gedanke, dem Benutzer, wenn er nicht auf die Kontrolleinheit sieht, so wenig optische Reize wie möglich zu geben, damit der Fokus nicht vom Hörerlebnis abgelenkt wird.

Für die DMX-Steuerung wurde ein Interface der Firma "Enttec" (DMX USB Pro) verwendet, welches eine Kommunikation via USB ermöglicht. Dazugehörig wurde das Max/MSP-Objekt `dmxusbpro` der Firma Nullmedium² benutzt, um direkt aus Max/MSP Daten schicken zu können. Die Lichtsteuerung folgt, wie Eingangs erwähnt, dem aktuellen Abspieltempo des Stückes. Die BPM-Zahl wird abgegriffen und je nach deren Wert in Bereichen von 20 BPM einer Farbe zugeordnet. Die Grenzen liegen nach unten hin bei 30 BPM, nach oben hin werden alle Werte über 180 BPM derselben Farbe zugeordnet. Die Übergänge zwischen den Farbbereichen geschehen fließend innerhalb von zwei Sekunden.

Wie bereits erwähnt, geschieht sämtliche grafische Umsetzung innerhalb von Max/MSP. Als Bildschirm und Kontrolloberfläche dient ein 24-Touch LCD der Firma "Ilyama" (ProLite B2483HS-B1).

C. Leap Motion

Durch den konstanten, technologischen Fortschritt entstehen laufend neue Wege zur Human-Computer-Interaction. Die auf diese Art entwickelten Schnittstellen eröffnen dem Nutzer viele Möglichkeiten. So verschieden die Technologien, so verschieden sind auch die Steuerungsdaten, welche für die Interaktion verwendet werden können. Zusätzlich zu haptischer

²<http://www.nullmedium.de>, 04.09.2017

und Sprachsteuerung können auch Kamerasysteme Daten ermitteln. Zwei Produkte, die oft für das Tracking von Körper- und Bewegungsdaten benutzt werden, sind die Xbox Kinect und die Leap Motion Kamera. Nach ausführlichen Vergleichen der Spezifikationen der zwei Kameras wurde aufgrund der besseren Einsatzfähigkeit im vorliegenden Kontext die Leap Motion Kamera verwendet.

Die Leap Motion Kamera kam 2013 mit dem Ziel auf den Markt, kleinste Bewegungen in sehr geringer Entfernung erfassen zu können. Während die Xbox Kinect Daten in einem sehr weiten Bereich von 0,5 - 3,5m abgreifen kann, ist der verwendbare Radius der Leap Motion mit 0,25 - 0,5m viel kleiner. Da dies jedoch für das Installationsdesign kein Problem darstellt, da nur das Tracking der Hände wichtig ist, fiel die Entscheidung zu Gunsten der Leap Motion Kamera. Auch die deutlich höhere Auflösung, welche Bewegungsänderungen in hundertstel Millimetern erfassen kann, galt als Entscheidungsgrundlage.

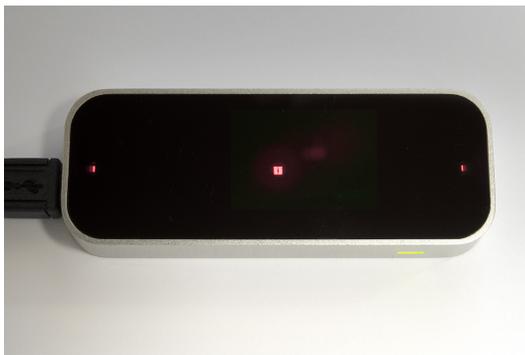
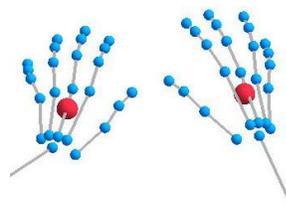


Abbildung 4. Leap Motion Kamera

Die Leap Motion Kamera funktioniert mittels drei LEDs, die das zu trackende Objekt beleuchten, welches durch zwei Infrarotkameras erfasst wird. Die Daten werden softwareseitig verarbeitet und in ein 3D-Modell der Hände umgewandelt. In der Software vorgegebene Bewegungsabläufe machen es möglich, verschiedene Bewegungen als Gesten zu erkennen (Swipe, Circle) oder das Tracking auf beliebige Weise einzuschränken (z.B. Tracking eines Fingers).

Die Tracking-Daten werden durch die eigene Software der Leap Motion Kamera erfasst. Mit Hilfe des Leap Motion Software Developer Kit und zweier Erweiterungen für Max/MSP konnten die Bewegungskordinaten in Max abgegriffen werden. In Isidor wurde aufgrund besserer Isolationsmöglichkeiten die Leap Motion Erweiterung des "IRCAM" (Institut de Recherche et Coordination Acoustique/Musique) benutzt.

Nach einigen Versuchen stellte sich heraus, dass die beste Methode für die Umwandlung von Dirigatsgesten das Tracken der Palm-Position in der Höhe (X-Achse) ist. Um nur diese Daten zu erhalten, wird der Datenstrom, in dem alle Daten vorhanden sind, gefiltert und nur die extrahierten Daten weiterverarbeitet.



palmwidth	86.57			
basis	0.78	-0.38	-0.49	0.59
direction	-0.19	0.61	-0.77	
palmposition	92.04	178.40	-7.14	
palmvelocity	0.25	-0.06	-0.01	
palmnormal	-0.99	-0.70	-0.40	
wristposition	108.67	145.67	29.76	
elbowposition	213.80	-23.76	191.51	
spherecenter	-5.58	73.84	-82.53	
sphereradius	166.09			
translationpro	0.33			
translation	0.00	0.00	0.00	
scaleprobability	0.39			
scalefactor	1.00			
rotationproba	0.28			
rotationangles	0.00	0.00	0.00	

Abbildung 5. Leap Motion Gesten-Tracking

Mithilfe eines Wahlselektors kann in der GUI bestimmt werden, welche Hand getrackt werden soll. Die so erhaltenen Daten werden nun interpretiert und in ein Tempo (BPM) umgewandelt, das, wie bereits erwähnt, als Multiplikator für die Masterclock weiterverwendet wird.

Zuerst werden erfasste Bewegungsdaten der X-Achse, welche sich zwischen 0 und 400 befinden, auf einen kleineren Wert (0-1) skaliert. Da das Projekt ursprünglich für Mac OS angedacht war, wurde in der Ursprungsversion für das Feststellen einer Richtungsänderung der Palm Position (oben/unten, entsprechend der klassischen Dirigatsgesten) das J.Delta-Objekt aus der Jamoma Library verwendet, das grundsätzlich die ersten drei Ableitungen einer Funktion berechnet. Durch den Umstieg auf ein Windows-Betriebssystem wurde dies jedoch aufgrund der fehlenden Implementierung unmöglich.

Um das J.Delta-Objekt zu ersetzen, werden die skalierten Daten in ein tfb- (Trigger-Float-Bang) Objekt geschickt, welches jede neue Koordinate speichert. Bekommt das Objekt einen neuen Input werden die beiden Werte subtrahiert und somit verglichen. Ist das Ergebnis negativ, findet eine Richtungsänderung statt. Jede negative Zahl wird an ein Change-Objekt weitergegeben, welches dafür sorgt, dass keine Koordinate doppelt verwendet wird. Jeder Negativwert löst nun einen Bang aus, der in ein Timer-Objekt geschickt wird, welches die vergangene Zeit zwischen zwei Bangs misst. Um von einem Zeitwert auf einen BPM-Wert zu kommen, wird das Ergebnis des Timer-Objekts durch 60000 dividiert. Um zu extreme Temposchwankungen zu verhindern, wird der mögliche BPM-Bereich auf 30-200 Schläge pro Minute festgelegt. Ein Slide-Objekt sorgt dafür, dass zwischen den Tempowerten interpoliert wird, was zu einem kontinuierlichen Tempowechsel führt. Um abschließend einen ganzzahligen BPM-Wert zu erhalten wird die errechnete Tempoinformation mit einem Round-Objekt gerundet.

Herausforderungen in der Programmierung stellte das Ersetzen des J.Delta-Objektes in der Windowsumgebung dar. Eine gute Übertragbarkeit der dirigierten Tempoinformation, um ein möglichst realistisches und steuerbares Erlebnis für den Nutzer zu ermöglichen, ist aufgrund des subjektiven Empfindens und dessen Ausdruck, anhand von Sensibilität des Trackings bzw. Feststellung des Zeitpunktes des tatsächlichen "Schlages", sehr schwer erreichbar.

VI. TESTLAUF

Um den Prototypen der Installation echten Rezipienten zum Test zur Verfügung zu stellen, wurde die Klanginstallation "Isidor" im Rahmen der "nextgeneration 7.0" im Zentrum für Kunst und Medien in Karlsruhe ausgestellt. Der Max/MSPPatch lief auf einem Macbook Pro (13z, Late 2011) auf dem Betriebssystem Windows 7. Als Audio-Interface wurde ein Saffire Liquid 56 benutzt, über dessen Outputs sechs Studiomonitore des Typs Genelec 1032B angesteuert wurden. Alle Filter an den Lautsprechern waren deaktiviert, die Gain-Regler waren alle auf den selben Wert eingestellt. Auch die Outputs des Saffire Liquid 56 spielten alle auf demselben Pegel aus; sämtliche Mischung geschah daher ausschließlich in Max/MSP. Als Lichter wurden zwei PAR56 LED-Scheinwerfer benutzt, die mit dem Laptop über das Enttec DMX USB Pro verbunden waren. Als Kontrolloberfläche diente der 24-Touchscreen der Firma Iiyama (Modell ProLite B2483HS-B1). Auf der Oberkante des Bildschirms lag die Leap Motion Kamera.

Beim Testaufbau wurde versucht, den Aufbaudimensionen (siehe Abb. 1) so weit wie möglich zu entsprechen. Aufgrund der Dimensionen des Raumes, in dem aufgebaut wurde, war dies jedoch nicht ausnahmslos möglich. Die beiden äußeren Frontlautsprecher standen in ca. 2 m Entfernung voneinander, der Zuhörer befand sich in ungefähr 2,5 m Entfernung von den Frontlautsprechern in mittiger Position. Die rückwertigen Lautsprecher standen innerhalb eines Meters zum Hörer, also sehr nahe im Vergleich zu den Idealdimensionen. Dieser Nachteil wurde durch zeitliche Verzögerung und Lautstärkenanpassung der Rear-Kanäle auszugleichen versucht.



Abbildung 6. Aufbau im Rahmen der "nextgeneration 7.0" im Zentrum für Kunst und Medien (ZKM) in Karlsruhe

Bei dem Raum, in dem der Testlauf durchgeführt wurde, handelte es sich um den Aufnahmerraum eines Tonstudios des ZKM, dementsprechend waren die akustischen Begebenheiten sehr günstig und die Charakteristik des Raumklanges als trocken zu bezeichnen. Der Raum hatte eine Gesamtfläche von ungefähr 25m² bei einer Höhe von rund 3 Metern.

Die Installation war zwei Tage lang zugänglich und konnte von Besuchern getestet werden. Die Versuchspersonen waren

gemischten Alters und Geschlechts und zu rund 70% Prozent Besucher der Ausstellung, während die restlichen 30% andere Ausstellende, Universitätsmitarbeiter oder Studenten des ZKM waren. Eine schriftliche Erklärung lag bereit; viele Besucher konnten dennoch anfängliche Berührungsängste mit der Installation erst nach einer persönlichen Einführung überwinden.

Die Reaktionen der rund 40 Probanden kann man als zum größten Teil positiv beschreiben. Viele Besucher schätzten vor allem die Möglichkeit der Temposteuerung durch das Dirigat, gaben aber auch an, dass sie sich noch nicht absolut in Kontrolle des Tempos fühlten. Alle Touch-gesteuerten Kontrollmöglichkeiten (Lautstärke, Position, Room, ...) wurden problemlos und mit Wohlwollen entgegengenommen. Auch das User Interface wurde als übersichtlich und optisch ansprechend empfunden. Die Autoren wurden von einem Probanden, der als Pädagoge arbeitet, auf das große Potenzial angesprochen, welches er in der Installation bezüglich Bildungsmöglichkeiten für Kinder sieht.

VII. HERAUSFORDERUNGEN

Die wohl größte Herausforderung ist es, ein für den Nutzer nachvollziehbares, kontrollierbares und spannendes Erlebnis zu kreieren. Ziel der Installation ist es, sowohl für Laien als auch für Personen mit fachspezifischem Hintergrundwissen ansprechend gestaltet und verständlich zu sein. Dazu ist es notwendig, dass alle Elemente reibungslos miteinander kommunizieren.

Dies ist die grundlegende Herausforderung für die Entwicklung der Installation. Es gilt, ein System zu schaffen, welches mehrere Elemente und Schnittstellen miteinander verbindet und eine Kommunikationsbasis schafft. Die Auswahl der zu verwendenden Technologien spielte dabei eine wichtige Rolle. Von Anfang an stellte sich die korrekte Detektion der Tempodaten und die darauffolgende Tempomanipulation als schwierig heraus. Hier ist es notwendig, eine feine Abstimmung zwischen der Technik und der Wahrnehmung des Rezipienten herzustellen. Für ein bestmögliches Nutzererlebnis sollte jeder Benutzer das Gefühl haben, er wäre in vollkommener Kontrolle über die Installation.

VIII. DISKUSSION

Beschrieben wurde eine Installation zur Steuerung eines virtuellen Streichquartetts mittels MIDI-gesteuerten VST-Instrumenten und Tracking von Dirigatsgesten mittels Leap Motion Kamera. Nach der Konzeption und einem Testlauf sowie Nachbesserungen kann gesagt werden, dass die Installation für den Benutzer in Bezug auf subjektives Kontrollempfinden, Handhabung und Vergnügen eine zufriedenstellende Erfahrung ist. Vor allem das klangliche Ergebnis durch den Einsatz von VST-Instrumenten und DSP ist aufgrund seiner Variabilität hinsichtlich Tempo, Anschlagsdynamik, Raumklang, etc. sehr befriedigend und bietet im Vergleich zu ähnlichen Installationen, welche mit aufgenommene Audiomaterial arbeiten, große Weiterentwicklungsmöglichkeiten. Der Benutzer kann mit einem leicht zu verstehenden Interface in viele Parameter eingreifen und bekommt umgehendes, audiovisuelles

Feedback. Entwicklungspotenzial gibt es nach wie vor in der Feinabstimmung, bspw. des Motion Trackings beim Dirigat. Da es sich bei der Installation um ein sehr reaktives System handelt, ist der Bedarf nach Feinjustierung naturgegeben sehr hoch, da jeder Eingriff viele verschiedene Parameter, vor allem hinsichtlich des Kontrollgefühls, beeinflusst.

IX. AUSBLICK

Möglichkeiten zur Weiterentwicklung des Projektes sehen die Autoren vor allem in der Leap Motion Technologie. Die Kamera erlaubt es dem Nutzer, beide Hände simultan zu erfassen und verschiedene Gesten zu erkennen. Die zweite Hand könnte somit dazu benutzt werden, andere Parameter anzusteuern - entsprechend dem Dirigat bspw. die Dynamik des Stückes. Wie bereits zuvor erwähnt, wurde von Probanden die Möglichkeit des Einsatzes der Installation für Bildungszwecke, speziell für Kinder, angemerkt. Letztlich bleibt auch die Weiterentwicklung des Bach-Objekts spannend für die Installation, da die Projektgruppe mit den Entwicklern von Bach in Verbindung trat und einige Anmerkungen und Anregungen möglicherweise in zukünftigen Versionen von Bach berücksichtigt werden könnten, was in der weiteren Entwicklung Wege ebnet würde. Um ein Beispiel zu nennen, würde die Möglichkeit, MIDI Control Changes in Bach verarbeiten zu können, große Verbesserungspotenziale für die realistische Klanggestaltung bieten.

Die Autoren hoffen, mit diesem Projekt Denkanstöße und Grundlagen für weiterführende Forschungen geschaffen zu haben.

LITERATUR

- [1] A. Jylhä, "Sonic gestures as input in human-computer interaction: Towards a systematic approach," in *Proceedings of the SMC2011-8th Sound and Music Computing Conference, Padova*, 2011.
- [2] E. Hornecker and M. Stifter, "Learning from interactive museum installations about interaction design for public settings," in *Proceedings of the 18th Australia conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments*. ACM, 2006, pp. 135–142.
- [3] A. Camurri, G. De Poli, and D. Rocchesso, "A taxonomy for sound and music computing," *Computer Music Journal*, vol. 19, no. 2, pp. 4–5, 1995.
- [4] J. O. Borchers, W. Samminger, and M. Mühlhäuser, "Conducting a realistic electronic orchestra," in *Proceedings of the 14th annual ACM symposium on User interface software and technology*. ACM, 2001, pp. 161–162.
- [5] WhiteVOID, "Mendelssohn Effektorium Conducting a virtual orchestra," Berlin, 2014, <http://www.whitevoid.com/#/main/interactivestructures/mendelssohn-effektorium>, letzter Zugriff: 04.09.2017.
- [6] Aconica, "Effektorium-Interactive Museography/Interactive Sound and Light Installation," Berlin, 2014, <http://www.aconica.de/portfoliopage/effektorium-interactive-museography/>, letzter Zugriff: 04.09.2017.
- [7] K. Flay, "Gesture controlled musical conducting," Semesterarbeit, School of Computer Sciences and Informatics Cardiff, Wales, 2015.
- [8] E. Sengpiel, "Intervall-Umrechnung in Frequenzverhältnis nach cent und zurück," n.d., <http://www.sengpielaudio.com/Rechner-centfrequenz.htm>, letzter Zugriff: 04.09.2017.