

**MIDI DÖNÜŞTÜRÜCÜ YAZILIMLARIN BAŞARI
KARŞILAŞTIRMASI VE
MATLAB'DA MÜZİK ANALİZİ (*)**

**Performance Comparison of MIDI Converting Software and
Music Analysis in MatLab**

DOI NO: 10.5578/amrj.67094

Cihan IŞIKHAN¹

Özet

Sanal ortamda müzik analizi, müzikle teknolojinin içi içe olduğu; birinin diğerini tamamladığı özellikli bir çalışma alanıdır. Analizde amaçlanan, yaratının dinler kitle üzerindeki sosyal bilimsel etkilerini araştırmak olmakla birlikte asıl olan, insan olanağıyla çözümlenemez verileri ortaya dökülebilmektir. Bu amaçla, son zamanlarda müzik alanında her türlü amaç için müzik verilerini hesaplanabilir sayısal değerlere dönüştürmesiyle müzisyenler arasında bile kullanımı giderek yaygınlaşan MatLab, müzik analizinde en etkili yazılımlardan biri haline dönüşmüştür.

Matlab ile müzik analizi ham ses dosyaları üzerinden yapılabildiği gibi daha yaygın olarak kullanılan yöntem, müziği sembolik verilere dönüştürerek işlemler süresince eldeki bu semboller üzerinden gitmektir. Böylesine güçlü ve bilgisayar hesaplamalarında kolaylık sağlayacak temsil gücü yüksek müziksel sembollerden biri MIDI'dir. Matlab, MIDI kullanarak çeşitli toolbox'larla (yardımcı araçlarla) müzik analizi yapabilir. Bu araçlardan biri, Finlandiya bilim insanları tarafından geliştirilen MIDI ToolBox'dır (MTB). MTB, tamamen MIDI verilerine dönüştürülmüş müzik dosyalarını kullanır ve yüze yakın sayıdaki analiz fonksiyonlarıyla veri işler. İşte tam bu noktada önemli olan, bir müzik dosyasını MIDI'ye dönüştürmektir.

Bu çalışmada, iyi bir MTB kullanımı için ses verilerini MIDI verilerine dönüştürebilen yazılımlar test edilmiştir. Klasik dönem bir piyano yarattığı birebir MIDI çevirisiyle teksesli ve çoksesli biçimde yazılımlarda kaynak olarak kullanılmış, elde edilen sonuçlar f-measure (f-ölçüm) yöntemiyle analiz edilmiştir. Sonuçlar göstermektedir ki, MIDI dosyası oluşturmak için en etkili çeviri yöntemi, doğrudan çalmak veya yazmaktır. Çünkü testte en başarılı yazılım olarak görülen WIDI ve Melodyn bile bugünkü haliyle hatalı veya eksik analizlere neden olabilir.

Anahtar Kelimeler: Müzik Analizi, MatLab, MIDI, Müzik Teknolojisi.

¹ Prof. Dr., Dokuz Eylül Üniversitesi, Güzel Sanatlar Fakültesi, Müzik Bilimleri Bölümü, Müzik Teknolojisi Anabilim Dalı, cihan.isikhan@deu.edu.tr

*Bu makale 10-12.05.2018 tarihleri arasında Kütahya'da düzenlenen "Müzik Teorileri" konulu IX. Hisarlı Ahmet Sempozyumu'nda sözlü bildiri olarak sunulmuştur.

Abstract

Computational music analysis is a specifically fields of study where music and technology entwined together and completing each other. The purpose of analysis, besides its social scientific influences, is to process some data that cannot be detected with human possibilities. MatLab, that becoming a comprehensive laboratory for all other engineering production and solutions and that converting music data into computable values for all purposes, has become one of the most effective software for music analysis.

Music analysis with Matlab can be applied by users over raw audio files, generally. But more commonly used method is to use symbolic data that represents to music. One of the strongest symbolic representations are MIDI. Matlab uses with a variety of toolboxes for music analysis such as MIDI Toolbox (MTB) that developed by Finnish scientists. Processing some data with hundreds of analysis functions, the MTB uses music files that have been completely converted to MIDI. At this point it is important to convert a music file to MIDI.

In this study, some software that convert audio to MIDI data has been tested. Monophonic and polyphonic form of piano music (sonata) in classical period have been used as a source of converting software, and the obtained results are analyzed by f-measure method. The results show that the most effective way to create a MIDI file is to play or write directly. Even WIDI and Melodyn, considered the most successful software in the test, may cause incorrect or incomplete analyzes with these results.

Keywords: *Music Analysis, MatLab, MIDI, Music Technology.*

GİRİŞ

Müzik teorisi, bir taraftan müziğin kuramsal çalışmalarıyla ilgili tüm sistematik alanları kapsarken, diğer taraftan analitik yaklaşımlarla müzikten veri toplama ve analiz etme uygulamalarını içerir. Kısaca müzik analizi olarak adlandırılan bu özel alanda yapılan çalışmalar, günümüz bilgisayar teknolojisiyle birlikte çok daha etkili bir disiplin haline gelmiştir. Bilgisayar tabanlı müzik analizi, basit gramer tabanlı tekniklerden (*simple grammar-based techniques*) dizi karşılaştırma ve arama tekniklerine kadar (*string matching and searching tehcniques*) birçok teknik ve yöntemle yürütülür ve bu çalışmalarda müzik, yalnızca karmaşık bir veri bütünüdür. Her müzik verisi, etkili veya değil (*data-poor/data-rich*) bir veri bankasında toplanabilir (McKay&Fujinaga 2007: 65). Kanadalı araştırmacı Huron, bilgisayar bilimleri ve müzik arasındaki ilişkiyi post-modern çağın sistematik ve doğal bir yapısı olarak nitelerken, kendi geliştirdiği ve *Humdrum* adını verdiği bir *toolkit* ile müzikten veri toplama çalışmalarına devam etmektedir (Huron 2002).

Veri bankası oluşturacak düzeyde müzikten veri toplamak, müziğe ait frekans, genlik, süre vs. gibi öznitelikleri tespit edip, ayrıştırıp onları sanal ortamda salt birer veri olarak depolamaktır. Bu veriler öyle güçlü ve doğru olmalıdır ki, koparılıp alındıkları yeri tam olarak temsil edebilsinler. Örneğin, manuel ortamda notalar müziği temsil eden yazılı birer veridir. Her nota uygulamaya geçildiğinde müziği ortaya çıkarır. Dolayısıyla notalar, müziği temsil eden güçlü birer veriye dönüşmüş olurlar. Bilgisayar bilimlerinin sanal ortamda aradığı işte budur. Ses kaydı yapılmış veya nota örneğinde olduğu gibi basılı hale getirilmiş bir müziğin içinden öznitelikler öyle alınıp sanal ortamda saklanmalıdır ki, geldiği bütünü temsil edebilecek derecede güçlü olsunlar.

Temsil gücü yüksek verilerden oluşmuş bir veri tabanı ile kullanıcılar, analiz ettiği müzikte perde (veya nota, frekans) sayılarını görmek isteyebilir. Bir veya ardışık birkaç perdenin müzikte kaç defa geldiğini sorgulayabilir. Perde veya nüansların tekrar sayısını öğrenmek isteyebilir. Melodideki seyri çizgisel görmek isteyebilir. Müziğin armonik yapısını incelemek, karakteristik bir takım unsurları saptamak isteyebilir. Analizini yaptığı müziğin diğer müziklerle veya aynı müziğin kendi içinde melodi, tema vs. benzerliğini sorgulayabilir. Kullanıcı tüm bunları çeşitli fonksiyonlar, algoritmalar veya yazılımlar aracılığıyla bir müziğin tonallığını veya makamını bulabilmek, bağdarın veya yaratının karakterini tespit edebilmek ve hatta birkaç notayla bilgisayara yaratı ısmarlamak gibi özgün buluşlara imza atmak amacıyla yapabilir. Dolayısıyla, müziği tam olarak temsil edebilen güçlü özniteliklerle oluşturulmuş bir müzik veri bankası, müzik analizi başta olmak üzere her türlü amaçla yapılacak işlemlerin doğruluğu açısından oldukça önemlidir.

Müzik analizi başta olmak üzere müzik verileriyle yapılan çalışmaların pek çoğunda, müzik özniteliklerinin ve dolayısıyla müziğin sembolik verilerle işleme alındığını, veri bankasında müziğin öznitelikleri gösterir çeşitli sembolik verilerle temsil edildiğini görmekteyiz. Örneğin Ghias ve ekibi perdeler arası yüksekliği gösterir *Parson* adlı özel bir sembolik veri kullanırken (Ghias 1995); Castan ve Good müziği temsil edebilecek en güçlü sembolik veri olarak XML'i (*eXtensible Markup Language*) işaret eder (Good 2000), (Castan 2001). Ancak Typke'nin günümüzde de geçerliliğini sürdüren araştırması, müziğin sembolik veri olarak en güçlü temsilcisinin MIDI olduğunu göstermektedir (Typke 2002).

MIDI (*Musical Instrument Digital Interface*), 1980'li yıllardan bu yana kullanılan ve halen etkisini sürdüren güçlü bir müzik formatıdır. MIDI'de müziksel öznitelikler sayısal kodlara

dönüştürülerek depolanır ve işlem görür. Dolayısıyla MIDI, bilgisayar bilimlerinde müziği temsil edebilir ve müzik analizinde kullanılabilir bir özellik taşır. Böylesine yaygın bir formatın müzik analizi vs. uygulamalarda kullanılması için çeşitli araçlar geliştirilmiştir. Kullanıcılar tarafından yaygın olarak kullanılan bu araçlardan biri, Eerola ve Toiviainen tarafından geliştirilen ve 2016 yılında tekrar revize edilerek kullanıma sunulan *MIDI ToolBox*'dır (Eerola&Toiviainen 2004, Eerola&Toiviainen 2016). Bu *ToolBox*, kullanıcılar için MatLab üzerinden işlem yapmayı sağlayan toplam 94 fonksiyon içerir. Aralarında MIDI dosyasını okutma, çalma ve düzeltme gibi temel olanlarının da bulunduğu fonksiyonlardan analiz amaçlı olan en önemlileri arasında ezgisel benzerliği ölçen “*meldistance*”, nota sayılarıyla işlem yapan “*nnotes*” ve anlamlı eşleşmeleri tespit edebilen “*segmentgestalt*” gösterilebilir. *ToolBox*, .mid formatla kayıtlı müzik dosyasını *readmidi* fonksiyonuyla okuyarak, tüm verileri *nmat* adlı bir matriste düzenler ve depolar. *nmat*, ağırlıklı olarak müzikteki perde, süre ve nüans bilgilerini içerir. Kullanıcı bu bilgilerle araçtaki 93 fonksiyonu (*readmidi* ile doksan dört) kullanarak aralarında analizlerin de bulunduğu çeşitli amaçlara yönelik sonuçları elde eder.

MIDI *ToolBox* gibi güçlü araçlar, beraberinde, kullanıcıları yakından ilgilendiren bir başka konuyu gündeme getirir: Ses dosyasından MIDI'ye dönüşüm. Bilindiği gibi her ne kadar müziği temsil etse de MIDI aslında yalnızca bir sembolik veridir, ses içermez. Oysa gerçek hayatta müzik için asıl olan ses dosyasıdır. Spesifik örnekler hariç müzikte istisnasız tüm kayıtlar, seslendirmeler, dinlemeler, performanslar vs. hep sesle yapılır, MIDI ile değil. O halde eğer gerçek hayatta MIDI yerine ses kullanılıyorsa, veri bankasına MIDI olarak depolanan müziklerin daha önce ses dosyasından MIDI verisine dönüştürülmesi gerekir. Dolayısıyla sesteki MIDI'ye dönüşüm oldukça önemlidir.

Bu çalışmada, özellikle Matlab MIDI *ToolBox* kullanılarak yapılacak tüm müzik analizleri için kullanıcıların daha hızlı çalışmaları açısından ses dosyalarını MIDI'ye çeviren yazılımların başarıları test edilmiştir. Eğer kullanıcılar ellerindeki ses dosyalarını MIDI'ye yazılımlar aracılığıyla çevirebilirlerse, *ToolBox* işlemlerine çok daha vakit ayırabilir, daha etkili analiz yapabilirler.

MIDI Dönüştürücü Yazılımların Başarı Karşılaştırması

Karşılaştırması yapılacak ve piyasada yaygın olarak kullanılan en ilk 14 yazılım tespit edilmiştir. Yazılımlar arasında herhangi bir ayırım yapılmamıştır ancak içlerinden WIDI gibi bazıları salt çevrim

aracı olarak üretilmişken, örneğin Cubase gibi olanlar çok amaçlı (*workstation*) olarak kullanılmaktadır. Bu türden bir sınıflandırmanın etkileri, elde edilen veriler doğrultusunda sonuç bölümünde tartışılacaktır. 14 yazılımın yalnızca süre kısıtlatmalı tam sürüm fonksiyonlu deneme sürümleri (*demo*), i5 işlemcili 8 GB RAM içeren Windows 10 işletim sistemli bilgisayara yüklenmiştir. Ancak 6 yazılım, yükleme sırasında veya sonrasında sürekli hata verdiği için işlem dışı bırakılmıştır. Böylece deneyde karşılaştırması yapılacak yazılım sayısı 8 (sekiz) olarak tespit edilmiştir.

Tablo 1: Karşılaştırmada tespit edilen ve kullanılan yazılımlar

	Tespit Edilen	Kullanılan
1	Ableton Live	Ableton Live
2	Akoff	Akoff
3	Amazing MIDI	Cubase
4	Composers	Digital Ear
5	Cubase	Intelli Score
6	Digital Ear	Logic Pro
7	Go Minimal	Melodyn
8	Intelli Score	Widi
9	Logic Pro	
10	Melodyn	
11	MIDI Editor	
12	Muse Book	
13	Neuratron	
14	Widi	

Karşılaştırılacak yazılımları test etmek amacıyla, çok sesli ve tek sesli olmak üzere 2 ayrı kategoride Şekil 1’de ilk sekiz ölçüsü sunulan Mozart’ın K 284 Piano D Majör Sonatı kullanılmıştır. Bu yaratının seçilmesindeki temel neden, klasik dönem diğer çoğu yaratıda olduğu gibi perdelerin inişli-çıkışlı, birbirleriyle matematiksel bir düzlem içindeymiş gibi seyretmesi olmuştur. Bu çizgi, yazılımların daha kolay işlem yapmasını sağlayacak bir kaynak oluşturmaktadır.

Şekil 1: Mozart K. 284 D Majör Piano Sonat, allegro (ilk 8 ölçü)

(a) Çoksesli (b) Teksesli (ezgi)

The image displays the musical score for the first 8 measures of Mozart's K. 284 D Major Piano Sonata, Allegro. Part (a) is the full piano score, showing the right and left hands with various dynamic markings such as *f*, *p*, and *cresc.*. Part (b) is the single melodic line (ezgi) in a single staff, showing the sequence of notes and rests.

Şekil 2: Ses dosyasının “widi” yazılımında MIDI’ye dönüşmesi

(a) Mzt.wav (b) Sonuç.mid

The image shows a screenshot of MIDI software. Part (a) displays the original .wav file with a piano roll showing the piano's performance. Part (b) shows the resulting .mid file with a piano roll and a MIDI editor, indicating the successful conversion of the audio file to a MIDI format.

Şekil 2’de gösterildiği gibi, müziğin .wav formatlı ses dosyası (Mzt.wav) yazılımlara *file>import* yolu kullanılarak aktarılmış,

dönüşüm sonucu oluşan mid uzantılı MIDI dosyaları, bilgisayarda sonuçlar klasörüne kaydedilmiştir.

Diğer taraftan, yaratının orijinaline sadık kalınarak hazırlanmış mid formatlı orijinal MIDI dosyası (Mzt.mid), farklı bir klasörde tutulmuştur. Sonuca ulaşmak, bir başka ifadeyle karşılaştırmaları yapabilmek için her bir yazılımın ürettiği sonuç mid dosyalarıyla Mzt.mid dosyasının MIDI *ToolBox*'ın “*meldistance*” fonksiyonuyla benzerlik ölçümü yapılmıştır. Bu fonksiyon, her iki MIDI dosyasının perdeleri arasındaki benzerliği sorgular. Bunun için, önce her iki dosya birleştirilerek tek bir .mid dosyası oluşturulur (karsilastirmasonuc1.mid). Ardından, Mzt.mid'e ait ilk 8 ölçüyle (karsilastirmasonuc1.mid'in ilk 8 ölçüsü), sonuç.mid'in ilk 8 ölçüsü (karsilastirmasonuc1.mid'in 641-649 ölçüleri arası) birebir karşılaştırır. Bunun için fonksiyonun aşağıdaki kodu kullanılmaktadır:

```
nm=reftune ('karsilastirmasonuc1.mid');
ph{1} = onsetwindow(nm,0,8);
ph{2} = trim(onsetwindow(nm,641,649));
ph{3} = trim(onsetwindow(nm,9,17));
ph{4} = trim(onsetwindow(nm,650,658));
...
...
for i=1:4
    for j=1:4
        dst(i,j)=meldistance(ph{i},ph{j},...
            'contour',);
    end
```

Fonksiyon, eşleşen perdelerle 1, eşleşmeyene 0 atayarak tüm dosyayı tarar ve ölçü-ölçü 0 ile 1 arasında değer döndürür. 1 tam eşleşmeyi, 0 hiç eşleşmemeyi gösterir.

Tablo 2: *Meldistance* Fonksiyon Sonuç Örneği

		Sonuc1.mid			
		Ölçü	A2	B2	C2
Mzt.mid	A1	.87	.21	.09	.78
	B1	.20	1.00	.12	.01
	C1	.09	.21	.98	.03
	D1	.45	.21.	.35	1.00

Tablo 2’de gösterildiği gibi beklenen, eşleşen perdeler nedeniyle aynı olan ölçülerin 1.00 döndürmesidir. Şekle göre B ve D ölçüleri 1.00 ile bunu sağlarken, A ve C ölçülerinde 1.00 yerine .87 ve .98 değerler görülüyor. Bunun anlamı, MIDI’ye dönüşümün başarılı olmadığı, perdelerin ilgili yazılım tarafından ıskalandığı veya doğru tespit edilememiş olmasıdır.

Karşılaştırma Sonuçları

Perde karşılaştırmalarının ardından, istatistiksel sonuçları ortaya koymak için *f-measure* tekniği kullanılmıştır. *F-measure*, karşılaştırılan iki verinin doğru saptanma oranını belirten bir ölçümdür. Oran ne kadar yüksekse başarı o derece yüksektir. *F-measure* (F) bu sonucu elde edebilmek için 2 farklı hesaplama yapar: Karşılaştırmanın kesinlik oranı (*Precision*, P) ve başarı oranı (*Recall*, R). Her iki hesaplamanın *f-measure* eşitliğinde formülü şöyledir:

$$F = \frac{2PR}{P + R}$$

Burada P ve R hesaplamaları, karşılaştırmada saptanan veya saptanamayan verilerin (notaların, perdelerin) doğruluk durumlarına göre yapılır. Önce, saptananlar pozitif (P), saptanmayanlar negatif (N) olarak tespit edilir. Ardından pozitiflerin içinde doğru (TP, *True Positive*) ve yanlış (FP, *False Positive*) olanlarla; negatiflerin içinde doğru (TN, *True Negative*) ve yanlış (FN, *False Negative*) olanların oranları hesaplanır. Bu işlemlerle aslında sorulan ve yanıtı beklenen ifade şudur: Nota, yazılım tarafından saptandıysa doğru mu saptanmış yoksa yanlış mı? Yazılım tarafından saptanmayanlar varsa bunlar doğru olarak mı saptanmamış yoksa bunlar da mı yanlış? Bu durumda P ve R hesaplamaları şu formülle yapılır:

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN}$$

Her yazılım için karşılaştırmada önce çoksesli sonra tek sesli müzik kullanılmıştır. Tüm bu işlemlerin ardından başarı karşılaştırmalarının F, P ve R sonuçları Tablo 3’de, *f-measure* sonuç değerlerine göre en başarılı yazılımdan başarısız doğru yapılan sıralama Tablo 4’de gösterilmiştir.

Tablo 3: Yazılımların F-Measure Sonuçları

ÇOKSESLİ	F-measure	Precision	Recall	TEKSESLİ	F-measure	Precision	Recall
Ableton Live	29.6%	29.1%	32.9%	Ableton Live	96.6%	96.1%	99.9%
Akoff	12%	8.5%	14.2%	Akoff	45%	41.5%	47.2%
Cubase	20%	16.5%	22.2%	Cubase	70%	66.5%	72.2%
Digital Ear	5.6%	1%	7%	Digital Ear	13.6%	9%	15%
Intelli Score	23.6%	23.1%	26.9%	Intelli Score	60.6%	60.1%	63.9%
Logic Pro	35%	31.5%	37.2%	Logic Pro	80%	76.5%	82.2%
Melodyn	43.6%	39%	45%	Melodyn	76.6%	72%	78%
WIDI	63%	59.5%	65.2%	WIDI	89%	85.5%	87.2%

Tablo 4: Yazılımların F-Measure Sonuçlarına Göre Başarı Sıralaması

ÇOKSESLİ	F-measure	Precision	Recall	TEKSESLİ	F-measure	Precision	Recall
WIDI	63%	59.5%	65.2%	Ableton Live	96.6%	96.1%	99.9%
Melodyn	43.6%	39%	45%	WIDI	89%	85.5%	87.2%
Logic Pro	35%	31.5%	37.2%	Logic Pro	80%	76.5%	82.2%
Ableton Live	29.6%	29.1%	32.9%	Melodyn	76.6%	72%	78%
Cubase	20%	16.5%	22.2%	Cubase	70%	66.5%	72.2%
Intelli Score	23.6%	23.1%	26.9%	Intelli Score	60.6%	60.1%	63.9%
Akoff	12%	8.5%	14.2%	Akoff	45%	41.5%	47.2%
Digital Ear	5.6%	1%	7%	Digital Ear	13.6%	9%	15%

Tablo 5: Sesten MIDI'ye Dönüştürücü Yazılımlarının Genel Başarı Sıralaması

GENEL	Çoksesli	Teksesli	Çözümleme	Verimlilik (süre,sn)
WIDI	63%	85.5%	57%	95% (17)
Melodyn	43.6%	76.6%	97%	99% (4)
Logic Pro	35%	80%	90%	87% (21)
Ableton Live	29.6%	96.6%	91%	24% (203)
Cubase	20%	70%	93%	87% (21)
Intelli Score	23.6%	60.6%	5%	75% (37)
Akoff	12%	45%	0%	0%
Digital Ear	5.6%	13.6%	0%	0%

Elde edilen sonuçlarla birlikte, *f-measure* değerlerinin yanı sıra, kısaca “elle hata düzeltme başarısı” olarak tanımlanabilecek 2

farklı sorunun oransal yanıtları da aranmıştır: Çözümleme olasılığı ve hız. Çözümleme olasılığı, kullanıcının yazılım üzerinde hatalı perde tespiti sonrası onu elle düzeltmek için müdahalesini gösterir. Bunun için profesyonel bir kullanıcının yazılım üzerinde her yaptığı veya yapamadığı eylemin sayısı ile süresi tarafımızdan tutulmuş ve kaydedilmiştir. Ardından elde edilen sayısal değerler yüzdelik oranlara dönüştürülerek Tablo 5 oluşturulmuştur. Tablo 5'e göre, tek sesli çevrimde en başarılı görülen Ableton Live'in sorun çözümlemede rakiplerine göre daha başarısız olduğu görülmektedir. Bunun aksine, başarı karşılaştırmasında alt sıralarda bulunan Cubase'in sorun çözmedeki oranının nispeten yüksek olduğu gözlemlenmiştir.

Her iki tabloda, tek sesli çevrimde Ableton Live'in, çok seslide WIDI'nin başarılı olduğu görülmektedir. Bu sonuçlara göre aslında, MIDI dönüştürmede en başarılı yazılım olarak WIDI'yi gösterebiliriz. Gerçek hayatta bir ses dosyasının daha çok çok sesli olacağı düşünülürse, bu durumda çevrim yazılımı olarak çok seslide ilk iki sırayı alan WIDI ve Melodyn kullanıcılara tavsiye edilebilir.

SONUÇ

Müzikte analiz, günümüzde bilgisayar aracılığıyla çok daha hızlı ve etkili işlem yapılabilen önemli bir çalışma alanıdır. Bu amaçla halen çok çeşitli *ToolBox*'lar geliştirilmektedir. Her *ToolBox*, müzikte etkili sonuç döndürme ilkesiyle hareket eder. Bu tür *ToolBox*'lardan biri de Finlandiyalı bilim insanları Eerola ve Toiviainen tarafından geliştirilen MIDI *ToolBox*'dır (Eerola&Toiviainen 2004). MIDI *ToolBox*, içeriğindeki 94 fonksiyonla tüm işlemleri MatLab üzerinde yapar ve tüm bu işlemler için MIDI formatlı müzik dosyalarına ihtiyaç duyar. Bu durumda bir kullanıcı için en büyük ihtiyaç, gündelik hayatta ses dosyası olarak bulunan müziklerin MIDI formata dönüştürülme işidir.

Bu çalışmada, piyasada ses dosyasından MIDI formata dönüşüm yapabilen 14 yazılım tespit edilmiş, bunların 8 tanesinin başarı karşılaştırması yapılmıştır. Bunun için Mozart'ın K. 284, 6 numaralı D majör piyano sonatının allegro bölümü hem çok sesli (orijinal haliyle) hem tek sesli (ezgi çizgisi) olarak 8 yazılımda MIDI formata dönüştürülmüş ve dönüşen müziğin orijinaline ne kadar benzediği *f-measure* tekniğiyle ölçülmüştür.

Karşılaştırmada en başarılı yazılımlar çok seslide %63 oranla WIDI, tek seslide %96.6 oranla Ableton Live olmuştur. Tek seslide ikinci sırayı WIDI aldığından, karşılaştırmalar galibinin WIDI olduğu söylenebilir. Ancak ne var ki WIDI salt MIDI dönüşüm amacıyla

üretilmesine rağmen hatalı dönüşümlere düzeltmede çok sınırlı bir ara yüze sahiptir. Başarıları nispeten düşük olmasına rağmen bu bağlamda en etkili yazılımlar Melodyn, Logic Pro ve Cubase'dir. Oysa Ableton Live dâhil söz konusu dört yazılımın öncelikli amacı MIDI dönüşüm değildir. Bu yazılımlar müzik endüstrisinde MIDI içerikli profesyonel ses editörleri (*workstation*) olarak bilinirler. Dolayısıyla WIDI'nin bu yazılımlara göre daha güçlü bir çevrim yapması ve ara yüzünü daha etkili hata düzeltme uygulamalarına ayırması beklenebilir.

Sonuç olarak, henüz müzik endüstrisinde tam olarak ses dosyasını MIDI formata çevirebilen bir yazılım üretilememiştir. Günümüz dönüşümde henüz en etkili çözüm, MIDI kullanan profesyonel yazılımlarda müziği doğrudan çalıp kaydederek veya notalarını yazarak MIDI'ye dönüştürmektir.

KAYNAKÇA

- Castan, G. (2001), *NIFFML: An XML Implementation of the Notation Interchange File Format*, Walter B. Hewlett&Eleanor Selfridge-Field (ed.), Cambridge, Massachusetts, MIT Press.
- Eerola T. ve Toiviainen P. (2004), "MIR in Matlab: The Midi Toolbox", *5th International Symposium of Music Information Retrieval*, Proceeding of ISMIR, pp. 22–27.
- Eerola T. ve Toiviainen P. (2016), *MIDI ToolBox for MATLAB, Manual Book*, Erişim (25.05.2018): <https://github.com/miditoolbox/>
- Ghias, A., Logan, J., Chamberlin, D. ve Smith, B.C. (1995), "Query by Humming: Musical Information Retrieval in an Audio Database", *ACM Multimedia'95*, Proceeding of ACM Multimedia, pp. 231–236.
- Good, M. (2000), "Representing Music Using XML", *2nd International Symposium of Music Information Retrieval*, Proceeding of ISMIR, pp. 49–57
- Huron, D. (2002), "Music Information Processing Using the Humdrum Toolkit: Concepts, Examples, and Lessons", *Computer Music Journal* 26 (2), pp. 11–26.
- McKay C. & Fujinaga I. (2007), "Style-Independent Computer-Assisted Exploratory Analysis of Large Music Collections", *Journal of Interdisciplinary Music Studies*, Spring 1 (1), pp. 63–85
- Typke, R., Wiering, F., Veltkamp, R.C. (2005), "A Survey of Music Information Retrieval Systems", *6th International Symposium of Music Information Retrieval*, Proceeding of ISMIR, pp. 153–1