# فصل سوم روش‌شناسی و پیاده‌سازی(دیتاست – پیش‌پردازش – استخراج ویژگی‌ها)

**روش‌شناسی و پیاده‌سازی(دیتاست– پیش‌پردازش – استخراج ویژگی‌ها)**هر نوازنده دارای یک "اثر انگشت" سبکی منحصر به فرد است که مانند امضای شخصی او عمل می‌کند. این ویژگی، بنیاد اصلی پژوهش حاضر است، بخش‌های پیاده‌سازی برای استخراج این ویژگی فردی برای هر دوی یک نوازنده‌ی تازه‌کار و آهنگساز به کار خواهد رفت تا بتوان به طور دقیق برای پژوهش بخش عملی درستی داشته باشیم. در این فصل، به جزئیات فنی و عملیاتی مدل پیشنهادی خود برای شبیه‌سازی و تحلیل این ارتباط می‌پردازیم. مدل ما به دنبال پاسخ به این پرسش کلیدی است: چگونه می‌توان با تحلیل دقیق ویژگی‌های نواختن، سبک یک نوازنده را به صورت محاسباتی شناسایی و با سبک‌های مختلف آهنگسازان تطبیق داد؟ این رویکرد، در نهایت به هر نوازنده کمک می‌کند تا نقطه‌ی شروعی متناسب با آمادگی سبکی درونی خود بیابد و مسیر یادگیری‌اش را بهینه‌سازی کند. این فصل، تشکیل دیتاست،‌ مراحل آماده‌سازی داده‌ها، و استخراج ویژگی‌ها را به تفصیل شرح می‌دهد.

#### ۳.۱. جمع‌آوری داده‌ها، مشکلات و بهینه‌سازی حافظه

**۳.۱.۱. جمع‌آوری داده‌ها**

پایه و اساس هر مدل یادگیری ماشینی، به‌ویژه در یک پژوهش پیچیده مانند این، بر کیفیت و دقتداده‌ها استوار است. برای ما، این امر از اهمیت ویژه‌ای برخوردار بود، چرا که مدل باید قادر به یادگیری ویژگی‌های ظریف و دشوار موسیقی و استخراج "اثر انگشت" متمایز هر آهنگساز باشد. از همین رو، یکی از حیاتی‌ترین مراحل، تهیه یک دیتاست بسیار دقیق و کاملاً متعادل بود.

برای این منظور، ما یک دیتاست ترکیبی ساختیم. هسته‌ی اصلی این مجموعه از GiantMidi-Piano به دست آمد؛ یک مجموعه داده از اجراهای زنده که شامل فایل‌های MIDI (Musical Instrument Digital Interface) است. این فایل‌ها نمایشی سمبولیک از نت‌ها، ریتم و دینامیک یک قطعه ارائه می‌دهند. MIDI به ما اجازه می‌دهد تا ساختار اساسی قطعات موسیقی را به صورت دقیق و بدون نویز صوتی تحلیل کنیم. با این حال، از آنجایی که هدف ما شبیه‌سازی شرایط واقعی نوازندگی بود که شامل ظرافت‌های صوتی و تکنیکی است، داده‌های MIDI را با داده‌های صوتی واقعی از اجراهای زنده تکمیل کردیم. این داده‌های صوتی از طریق کانال‌های مختلف یوتیوب و با استفاده از یک کد پایتون اختصاصی که بر پایه کتابخانه‌ی youtube-dl نوشته شده و در مخزن گیت‌هاب پیاده‌سازی شده بود، جمع‌آوری شدند. این مجموعه داده، حاصل سال‌ها تدریس و یادگیری پیانو است و به صورت یک فایل اکسل دسته‌بندی شده است.

#### ۳.۱.۲. نیاز به داده‌های تکمیلی و دستی

با وجود جامعیّت، دیتاست اولیه کاملاً کامل نبود. در طول تحلیل‌های اولیه، متوجه شدیم که برخی از قطعات و سبک‌های مهم آهنگسازان در آن وجود ندارد. این قطعات برای درک کامل "اثر انگشت" آن‌ها و همچنین برای آموزش مدل به یک درک جامع از سبک‌های مختلف، حیاتی بودند. به همین دلیل، تصمیم گرفتیم ۲۰۰۰ قطعه‌ی دیگر را به صورت دستی جمع‌آوری و به دیتاست اضافه کنیم. این کار، نه‌تنها تعداد آهنگ‌های دیتاست را به ۸۰۰۰ آهنگ متمایز از ۷۳ آهنگساز رساند، بلکه به ما اطمینان داد که داده‌ها نماینده‌ی کاملی از تنوع سبک‌ها و تکنیک‌های نوازندگی هستند. این مجموعه دستی، شامل قطعاتی است که به صورت خاص برای این پژوهش و بر اساس نیازهای مدل ما انتخاب شده‌اند و به این ترتیب، به عمق و غنای مجموعه داده‌ی ما افزودند. در نهایت، همه‌ی این داده‌ها، چه از منابع اصلی و چه از طریق جمع‌آوری دستی، در یک دیتاست واحد ادغام شدند و آماده‌ی مراحل بعدی پیش‌پردازش و تحلیل شدند. این فرآیند دقیق و زمان‌بر، تضمین کرد که پایه‌ی مدل ما بر روی داده‌هایی محکم و جامع بنا شده است.[(کد مربوط به استخراج دیتاست)](#dataset)

#### Screenshot 1404-06-09 at 11

شکل ۳-۱ فرمت اکسل طبقه‌بندی شده

#### ۳.۲. متعادل‌سازی دیتاست

متعادل‌سازی تعداد داده‌ها در عمل می‌تواند به روش‌های بی‌شماری صورت بگیرد. چیزی که در این مرحله از اهمیت بالایی برخوردار است، جلوگیری از گم شدن داده‌های اصلی در میان داده‌های اضافه شده برای یکسان‌سازی تعداد است. اگر این فرآیند به صورت ناآگاهانه و بی‌رویه انجام شود، می‌تواند دقت مدل آینده را به شدت کاهش دهد. برای جلوگیری از این موضوع، برای متعادل‌سازی دیتاست از دو روش اصلی کمک گرفتیم:

۳.۲.۱ افزایش داده با استفاده از آگمنتاسیون (Augmentation)

در یک دیتاست نامتوازن، مدل ممکن است به سمت کلاس‌هایی با حجم بیشتر سوگیری پیدا کند. برای حل این مشکل، می‌توان با استفاده از روش آگمنتاسیون(Data Augmentation)، به کلاس‌هایی که تعداد قطعه‌های کمتری دارند، داده اضافه کرد. این تغییر می‌تواند شامل اضافه کردن نویز، تغییر دادن تمپو، تغییر دادن گام قطعه، و همچنین شیفت دادن برخی از هارمونیک‌های قطعات باشد. نکته‌ی قابل توجه این است که برای این روش باید به تعداد داده‌ها دقت کرد. برای مثال، اگر آهنگسازی مانند باخ در دیتاست ما ۲۰۰۰ قطعه داشته باشد که ۱۰۰ قطعه‌ی آن با آگمنتاسیون به دست آمده باشد، کلاس‌بندی و تشخیص ویژگی‌های او تفاوت بسیاری با آهنگسازی خواهد داشت که ۱۰۰ قطعه‌ی اصلی و ۲۰۰۰ قطعه‌ی دستی ساخته شده دارد. بنابراین، نیاز است که الگوریتمی برای مدیریت این موضوع ارائه شود.

الگوریتم ارائه شده برای این بخش به شرح زیر است:

1. ابتدا برای هر آهنگساز، تعداد قطعات شمارش می‌شود.
2. یک آستانه برای تعداد داده‌های هر آهنگساز تعریف می‌شود. هر آهنگسازی که کمتر از میانگین تعداد قطعات را داشته باشد، حذف می‌شود.
3. با توجه به آهنگسازان باقی‌مانده، میانه‌ی تعداد محاسبه می‌شود.
4. داده‌ها با توجه به فاصله‌ی تعداد قطعات از میانه، با ضریب ۱.۲ تا حداکثر ۱.۴ آگمنت می‌شوند.
5. بنابراین، برای کمترین تعداد قطعه در یک کلاس، ضریب ۱.۴ و برای بیشترین تعداد، ضریب ۱.۲ اعمال می‌شود. این روش باعث می‌شود که علاوه بر متعادل‌تر شدن دیتا، هیچ ویژگی‌ای به دلیل استفاده‌ی زیاد از روش، از دست نرود.

۳.۲.۲. متعادل‌سازی دیتاست با بخش‌بندی داده‌ها (Segmentation)

برای این که بتوان داده‌های موسیقی را به صورت درستی تحلیل کرد نیاز است که یک قطعه‌ی مثلا ۱۰ دقیقه‌ای به بخش‌های ۳۰ ثانیه‌ای تقسیم شود. این‌گونه در صورتی که بخشی شامل انرژی کم و یا ویژگی کم باشد حذف می‌شود و همینطور در نهایت تحلیل ویژگی‌ها بهتر صورت می‌گیرد. زمانی که داده‌ها متعادل‌تر باشند می‌توان از یک روش ثابت و با استفاده از یک زمان‌بندی ثابت برای هر قطعه استفاده کرد. اما به دلیل بالانس نبودن ما از یک الگوریتم پویا برای تقسیم‌بندی (segmentation) داده‌ها به قطعات زمانی کوتاه‌تر (سگمنت) استفاده کردیم.

1. ابتدا تمامی آهنگ‌ها به سگمنت‌های ۳۰ ثانیه‌ای تقسیم شدند.
2. سپس، تعداد سگمنت‌های هر آهنگساز شمارش شد.
3. اگر تعداد سگمنت‌های یک آهنگساز از میانگین ۳۰ آهنگساز با بیشترین تعداد سگمنت کمتر بود، طول سگمنت‌ها به صورت پویا کاهش می‌یافت (تا حداقل ۱۵ ثانیه) و شمارش دوباره انجام می‌شد.

این فرآیند ادامه یافت تا به تعادل نسبی برسیم. اگر با وجود این فرآیند، تعداد سگمنت‌ها همچنان زیر میانگین بود، آن آهنگساز از مجموعه کلاس‌ها حذف می‌شد. در نهایت، این روش منجر به ایجاد ۲۳ کلاس متعادل شد.

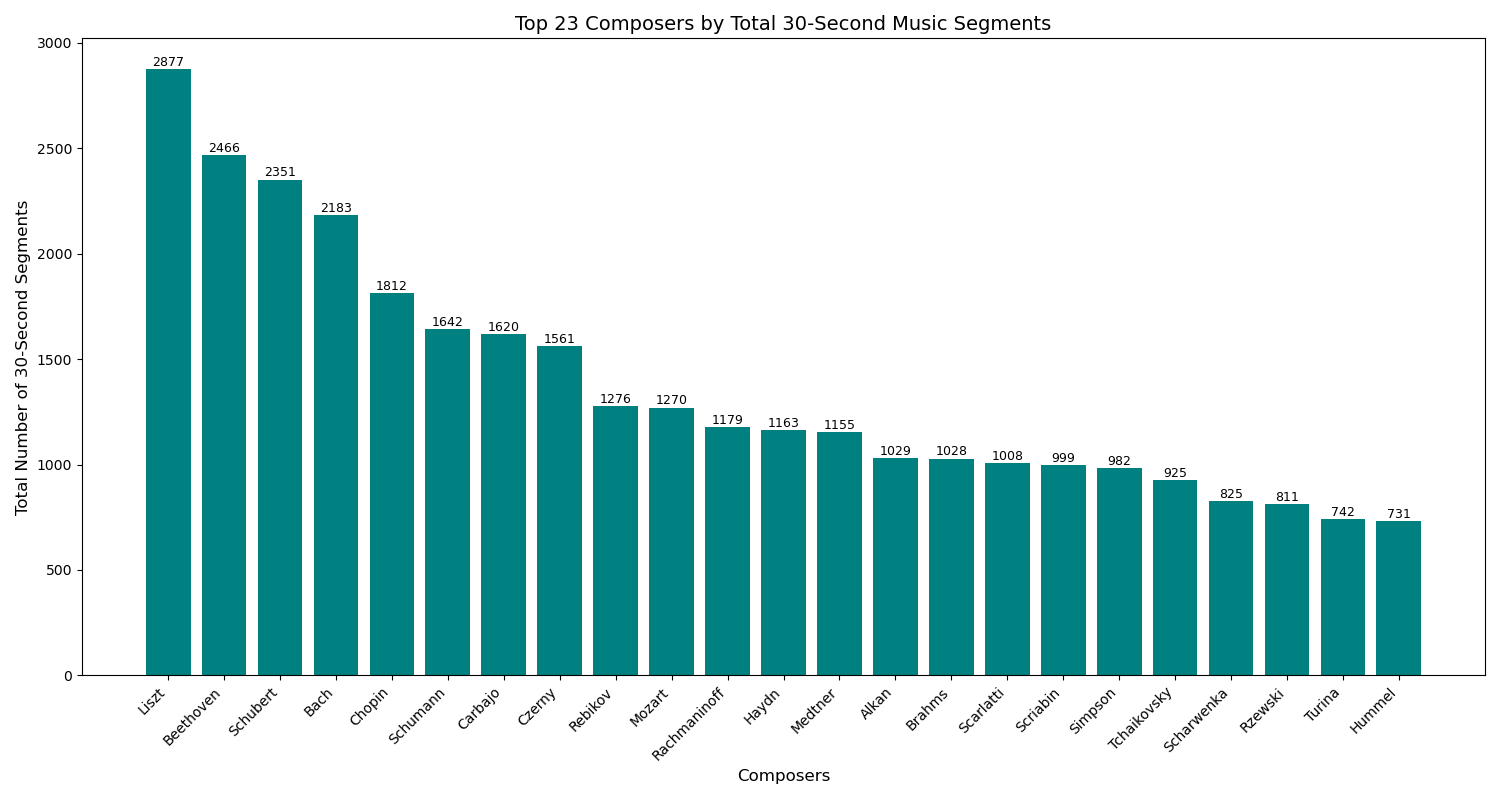
۳.۲.۳. مشکلات مدیریت حافظه برای بالانس کردن داده‌ها

زمانی که می‌خواهیم از الگوریتم‌های گفته شده برای یکسان کردن تعداد بخش‌ها استفاده کنیم، باید مدیریت حافظه‌ی کد را به خوبی انجام دهیم. در صورت عدم مدیریت صحیح، ممکن است با مشکلاتی مواجه شویم. به عنوان مثال، اگر برای شمارش تعداد سگمنت‌های تمامی قطعات، آن‌ها را داخل آرایه‌ها یا لیست‌های پایتون ذخیره کنیم یا همواره نتایج میانی را در RAM بنویسیم، در نهایت باعث به‌وجود آمدن مشکل اساسی خواهیم شد، و درنهایت این عوامل علاوه بر کند کردن اجرا، باعث خطای حافظه خواهند شد.

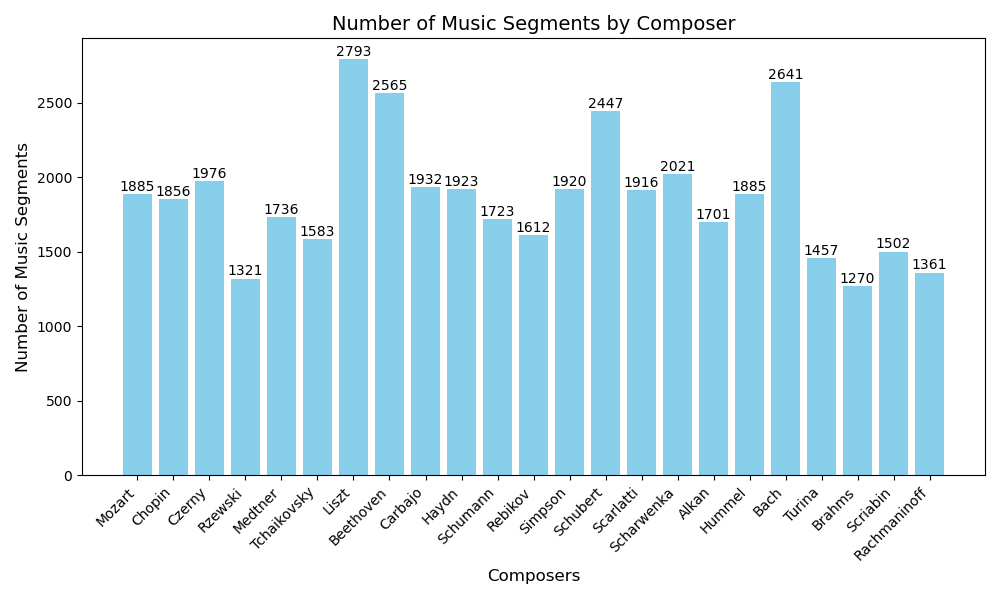
* **مدیریت حافظه (Memory Management):** همان‌طور که گفته شد، این فرآیند نیازمند پردازش و ذخیره‌سازی حجم عظیمی از داده‌ها است. با توجه به توضیحات، در طول عملیات همزمان خواندن و نوشتن، با خطای حافظه مواجه شدیم. برای رفع این مشکل، از چندین تکنیک بهینه‌سازی حافظه استفاده شد:
* **بارگذاری تدریجی داده‌ها (Lazy Loading):** به جای بارگذاری کل دیتاست در حافظه، داده‌ها به صورت دسته‌ای (batch) و فقط در زمان نیاز بارگذاری شدند. این کار باعث کاهش چشمگیر مصرف حافظه و جلوگیری از خطای سیستمی شد.
* **استفاده بهینه از ساختارهای داده:** به‌جای استفاده از لیست‌های پایتون که فضای زیادی در حافظه اشغال می‌کنند، از ساختارهای داده‌ای مانند آرایه‌های NumPy استفاده شد که به صورت بهینه‌تری داده‌ها را ذخیره می‌کنند.
* **مدیریت فایل‌های موقت:** برای ذخیره‌سازی نتایج میانی (مانند تعداد بخش‌ها پس از شمارش اولیه)، به جای نگهداری همه چیز در RAM، فایل‌های موقت ایجاد و پس از استفاده حذف شدند. این روش به خصوص برای بخش‌های حجیم فرآیند تقسیم‌بندی داده‌ها کارآمد بود.

۳.۲.۴. فرمت داده‌های استفاده شده در دیتاست

برای یک دیتاست حجیم که ورودی یک مدل یادگیری ماشینی یا الگوریتم‌های مشخص خواهد بود، نیاز به استفاده از یک فرمت مشخص، کم‌حجم و بدون افت کیفیت است. از این رو، برای ذخیره‌سازی داده‌ها از فرمت ‍".flac" استفاده شده است. این فرمت از دیتا ویژگی‌هایی دارد که برای پژوهش پیش رو بسیار مناسب است. ".flac" یک فرمت صوتی با فشرده‌سازی بی‌اتلاف (Lossless Compression) است. این به معنای آن است که کیفیت اصلی فایل صوتی بدون هیچ‌گونه افت کیفیتی حفظ می‌شود، در حالی که حجم آن نسبت به فرمت‌های غیرفشرده مانند"wav." کاهش می‌یابد. انتخاب این فرمت به ما اطمینان می‌دهد که هنگام تحلیل ویژگی‌های صوتی، هیچ‌گونه اطلاعات مهمی از دست نمی‌رود. این مسئله برای دقت مدل ما بسیار حیاتی است.



شکل ۲-۳ تعداد قطعات هر آهنگساز پیش از متعادل‌سازی



شکل ۳-۳ تعداد قطعات هر آهنگساز بعد از متعادل‌سازی

#### ۳.۲ . پیش‌پردازش داده‌ها (Pre-processing)

پس از جمع‌آوری و متعادل‌سازی دیتاسِت، مرحله‌ی بعدی پیش‌پردازش داده‌های موسیقی است. این فرآیند شامل چندین گام است تا داده‌ها را برای استخراج ویژگی و آموزش مدل آماده کند.

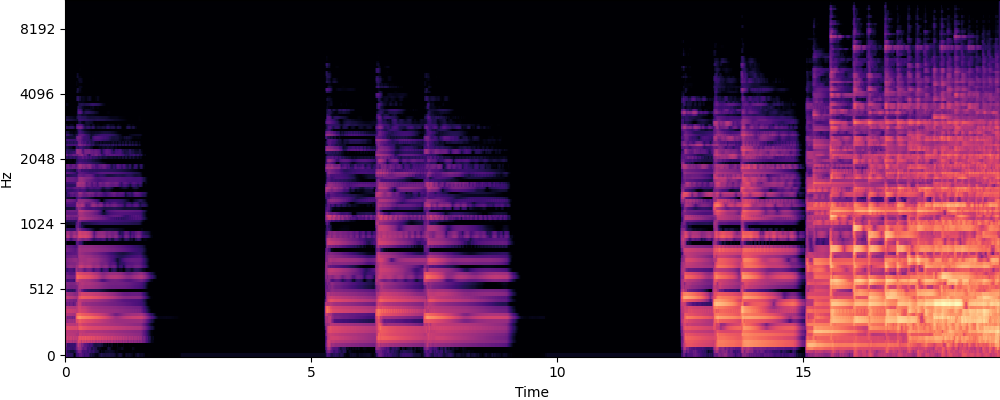
۳.۲.۱. حذف سگمنت‌های کم‌انرژی

یکی از مراحل حیاتی در پیش‌پردازش داده‌ها، غربالگری و حذف سگمنت‌های فاقد ارزش اطلاعاتی است. این سگمنت‌ها اغلب شامل سکوت‌های طولانی یا بخش‌هایی با انرژی صوتی بسیار پایین هستند. به‌طور خاص، سگمنت‌های ابتدایی یک قطعه موسیقی معمولاً علاوه بر انرژی صوتی کم، حاوی ویژگی‌های آهنگسازی اندکی هستند. نگهداری این داده‌ها در مجموعه آموزشی می‌تواند به دقت مدل آسیب برساند، زیرا مدل را وادار به یادگیری از روی داده‌های بی‌معنی و نویزدار می‌کند.

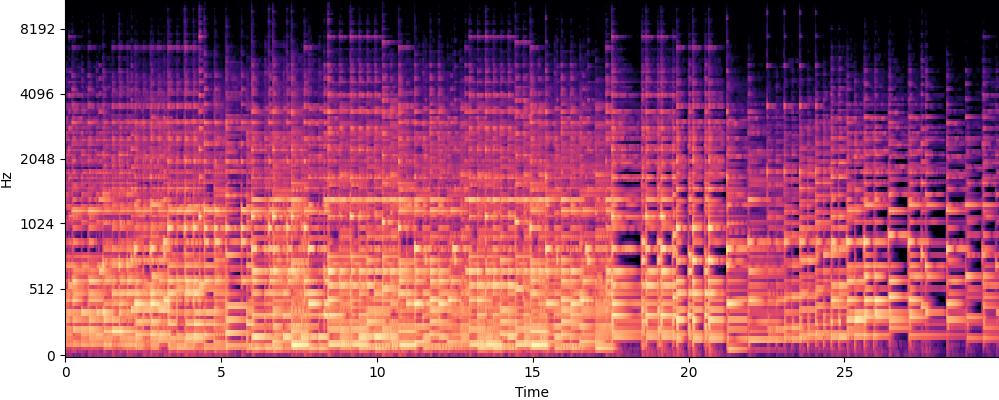
برای حل این چالش، سگمنت‌هایی که میانگین انرژی صوتی آن‌ها از یک آستانه‌ی مشخص کمتر بود، حذف شدند. این رویکرد به مدل امکان می‌دهد تا صرفاً بر روی داده‌های معنادار و غنی از اطلاعات موسیقایی تمرکز کند و در نتیجه، دقت طبقه‌بندی افزایش یابد.

* **فرآیند تعیین آستانه‌ی بهینه**: تعیین آستانه‌ی مناسب یک فرآیند حساس بود که از طریق آزمون و خطا صورت گرفت. این فرآیند با بررسی تأثیر آستانه‌های مختلف در بازه‌ی ۰.۰۰۰۱ تا ۰.۰۱ بر عملکرد مدل انجام شد:
* **آستانه‌ی بسیار پایین**: انتخاب یک آستانه‌ی بسیار پایین منجر به حذف حداقلی داده‌ها می‌شد. این امر باعث می‌شد اطلاعات غیرمفید و نویزدار در دیتاست باقی بماند که در نهایت دقت مدل را کاهش می‌داد، زیرا مدل در تمییز الگوهای واقعی از نویز با دشواری مواجه می‌شد.
* **آستانه‌ی بسیار بالا**: از سوی دیگر، انتخاب یک آستانه‌ی بسیار بالا ممکن بود به حذف بخش‌های مهمی از قطعه منجر شود که دارای انرژی کمتر اما اطلاعات موسیقایی ارزشمندی هستند (مانند بخش‌های آرام و ظریف). این امر موجب از دست رفتن داده‌های حیاتی برای تشخیص سبک و تکنیک‌های نوازندگی می‌شد و فرآیند یادگیری مدل را مختل می‌کرد.

پس از تحلیل نتایج متعدد، یک آستانه‌ی بهینه انتخاب شد که تعادل مناسبی میان حذف داده‌های بی‌فایده و حفظ اطلاعات کلیدی برقرار می‌کرد. این فرآیند دقیق، نقشی اساسی در بهبود کارایی کلی مدل ما ایفا می‌کند و تضمین می‌نماید که مدل بر روی باکیفیت‌ترین داده‌ها آموزش می‌بیند و توانایی آن در شناسایی "اثر انگشت" آهنگسازان به حداکثر می‌رسد.



شکل ۴-۳ تبدیل فوریه گسسته از یک سگمنت با انرژی پایین



شکل ۵-۳ تبدیل فوریه‌ گسسته از یک سگمنت با انرژی بالا

۳.۲.۲. نرمال‌سازی(Normalization)

برای جلوگیری از تأثیر دامنه (amplitude) متفاوت فایل‌های صوتی بر روی ویژگی‌ها، تمامی سگمنت‌ها نرمال‌سازی شدند. این فرآیند، دامنه‌ی سیگنال را به یک بازه‌ی استاندارد (مثلاً]۱+,۱-[) می‌آورد. در اصل دامنه‌ی سیگنال‌های صوتی بر روی ویژگی‌های فرکانسی آن‌ها بی تاثیر است، اما نکته‌ی حائز اهمیت آن است که در استخراج ویژگی‌های منحصر به یک آهنگساز علاوه بر عوامل فرکانسی، تفاوت‌های دامنه‌ای، هارمونیکی، و تیمبری (timbre) نیز اثر گذارند.

۳.۲.۳. بازنمونه‌برداری (Resampling)

نرخ نمونه‌برداری (Sample Rate)در اصل تعداد نمونه‌های صوتی گرفته شده در هر ثانیه است. این نرخ در فایل‌های صوتی مختلف می‌تواند متفاوت باشد. برای اطمینان از یکدست بودن دیتاست، تمام فایل‌ها باید به یک نرخ نمونه‌برداری واحد تبدیل شوند. این فرآیند که بازنمونه‌برداری نام دارد، داده‌ها را به یک استاندارد مشترک می‌رساند. برای مثال، اگر فایل‌ها با نرخ‌های ۴۴.۱ کیلوهرتز و ۴۸ کیلوهرتز وجود داشته باشند، می‌توان همه را به یک نرخ ثابت مانند ۴۴.۱ کیلوهرتز تبدیل کرد. این کار به الگوریتم‌های پردازش سیگنال و مدل‌های یادگیری عمیق اجازه می‌دهد تا با داده‌های یکدست و قابل مقایسه کار کنند. اگر این کار انجام نشود، ممکن است نتایج مدل دچار خطا شوند، چرا که مدل، ویژگی‌های استخراج شده از نرخ‌های نمونه‌برداری متفاوت را به اشتباه تفسیر می‌کند.[(کد نهایی مربوط به پیش‌پردازش)](#preprocessing)

#### ۳.۳. انواع ویژگی‌ها و استخراج آن‌ها

برای اینکه مدل بتواند قطعات موسیقی را کلاس‌بندی کند، نیاز به تبدیل سیگنال صوتی به بردارهای ویژگی‌های عددی داریم. این ویژگی‌ها باید اطلاعات کلیدی مانند هارمونی، ریتم و دینامیک را در خود داشته باشند. در این پروژه، از سه ویژگی اصلی استفاده کردیم و آن‌ها را به عنوان ورودی مدل قرار دادیم. در ضمن موضوع حائز اهمیت، جامعیت ویژگی‌ها است. اگر ویژگی‌های استخراج شده نتوانند تمامی ابعاد یک موسیقی را بررسی کنند، کلاس‌بندی نهایی به درستی و با در نظر گرفتن تمامی شرایط صورت نخواهد گرفت. بنابراین پس از توضیحات مربوط به هر ویژگی، منحصرا به لزوم استفاده از هر کدام و جامعیت آن‌ها باهم دیگر می‌پردازیم.

۳.۳.۱. تبدیل مل اسپکتروگرام (Mel Spectrogram)

مل اسپکتروگرام یک نمایش بصری از محتوای فرکانسی سیگنال صوتی در گذر زمان است. این ویژگی، مقیاس فرکانسی را به مقیاس Mel تبدیل می‌کند که به شیوه‌ی درک گوش انسان از فرکانس‌ها نزدیک‌تر است. برای درک عمیق‌تر مل-اسپکتروگرام، ابتدا باید با مقیاس مل آشنا شویم. مقیاس مل (Mel Scale)، یک مقیاس ادراکی (perceptual) برای اندازه‌گیری زیر و بمی صداست. این مقیاس توسط استیونز، ولکمن و نیومن در سال ۱۹۳۷ ابداع شد و نام آن از کلمه‌ی "Melody" (ملودی) گرفته شده است، زیرا هدف آن ایجاد مقیاسی بود که زیر و بمی‌های برابر را به فواصل برابر درک شده توسط انسان مرتبط کند[[۳۸]](#link38). در واقع، این مقیاس تلاش می‌کند تا پاسخی به این حقیقت بیولوژیکی بدهد که گوش انسان به فرکانس‌های مختلف به شکل متفاوتی واکنش نشان می‌دهد. گوش ما به تغییرات در فرکانس‌های پایین بسیار حساس‌تر است و با افزایش فرکانس، حساسیت آن به تغییرات کاهش می‌یابد. به عنوان مثال، تفاوت بین ۲۰ هرتز و ۴۰ هرتز برای ما بسیار محسوس‌تر از تفاوت بین ۱۰۰۰۰ هرتز و ۱۰۰۲۰ هرتز است، حتی با وجود اینکه در هر دو حالت، تفاوت فرکانس فقط ۲۰ هرتز است [[۳۹]](#link39). در نهایت نیاز از که یک مقایسه میان مقیاس خطی فرکانس و مقیاس مل داشته باشیم :

* **مقیاس خطی(هرتز) :** در این مقیاس، فاصله‌ی بین فرکانس‌ها ثابت است. این مقیاس برای ابزارهای فیزیکی و تحلیل‌های ریاضی مناسب است، اما با نحوه‌ی درک انسان از صدا مطابقت ندارد.
* **مقیاس مل** : این مقیاس غیرخطی است. در فرکانس‌های پایین، مقیاس مل به صورت خطی با هرتز مطابقت دارد، اما در فرکانس‌های بالا، رابطه به صورت لگاریتمی تغییر می‌کند. این بدان معناست که برای دستیابی به یک "فاصله‌ی زیر و بمی" ثابت از نظر ادراکی، نیاز به یک تغییر فرکانس بسیار بزرگتر در هرتز داریم[[۴۰]](#link40).

در نهایت باید بدانیم که برای تبدیل مقیاس خطی به مقیاس مل نیاز به فرمول زیر است :

* m در اینجا مقدار در مقیاس مل و f مقدار در هرتز است.

علاوه بر ویژگی‌های گفته شده،‌ نیاز است بدانیم که چرا استفاده از مل اسپکتروگرام که از مقیاس مل بدست می‌آید، کاربردی است :

* **هم‌سویی با ادراک انسانی :** با توجه به اینکه مقیاس مل نحوه‌ی شنیدن انسان را شبیه‌سازی می‌کند، ویژگی‌های استخراج‌شده با این مقیاس برای مدل نیز اهمیت بیشتری پیدا می‌کنند. این کار به مدل اجازه می‌دهد تا بر روی ویژگی‌هایی تمرکز کند که برای گوش انسان در تشخیص سبک و زیر و بمی مهم هستند.
* **کاهش ابعاد و فشرده‌سازی اطلاعات :** مل-اسپکتروگرام با تبدیل یک سیگنال صوتی بسیار پیچیده و حجیم به یک تصویر نسبتاً کوچک، باعث کاهش ابعاد داده‌ها می‌شود. این فشرده‌سازی اطلاعات باعث می‌شود که مدل بتواند با داده‌های کم‌حجم‌تر و با سرعت بیشتری آموزش ببیند. در نهایت لازم است در نظر داشته باشیم که حتی با توجه به کم حجم بودن اطلاعاتی که مل-اسپکتروگرام شامل می‌شود،‌ اما در نهایت دارای ارزش بسیار زیاد از نظر یک مدل یادگیری ماشین است و شامل پارامترهای زیادی می‌شود.
* **مناسب برای مدل‌های پردازش تصویر :** از آنجایی که خروجی مل-اسپکتروگرام یک ماتریس دوبعدی است، می‌توان آن را به عنوان ورودی به مدل‌های قدرتمند پردازش تصویر مانند CNN داد. این مدل‌ها به طور ذاتی برای شناسایی الگوهای فضایی (مانند الگوهای رنگی در تصویر مل-اسپکتروگرام) بهینه شده‌اند و می‌توانند به بهترین شکل ویژگی‌های پنهان سبک‌های موسیقی را کشف کنند. در نهایت از نظر پردازش تصویر مدل‌های قدرتمندتری نیز در این موضوع به کار می‌روند، این مدل‌ها که اکثرا از پیش به روی دیتاست های بزرگ تصویر اعمال شده‌اند و از پیش، بهترین وزن‌های مربوط به داده‌های تصویری را دارا می‌باشند. مدل‌هایی که برای تصاویر استفاده می‌شوند، در لایه‌های ابتدایی ویژگی‌های ساده‌تر مانند لبه‌ها و خطوط را درک می‌کنند، که در اصل بخش اصلی این تبدیل‌ها نیز در لایه‌های ابتدایی شناسایی خواهند شد. به این موضوع دوباره در قسمت مدل‌ها اشاره خواهد شد.

در نهایت برای این تبدیل نیاز است بخش فنی آن برای استخراج ویژگی نیز مورد بررسی قرار بگیرد. فرایند استخراج مل-اسپکتروگرام از یک سیگنال صوتی، شامل چند گام کلیدی است :

* **تبدیل فوریه زمان کوتاه (STFT) :** ابتدا سیگنال صوتی به فریم‌های کوتاه و همپوشان تقسیم می‌شود. روی هر فریم، تبدیل فوریه اعمال می‌گردد تا محتوای فرکانسی آن فریم به دست آید. خروجی این مرحله یک اسپکتروگرام با مقیاس فرکانس خطی است.
* **اعمال فیلتربانک مل (Mel Filterbank) :** خروجی مرحله قبل، از میان یک سری فیلتر مثلثی شکل عبور داده می‌شود که به صورت لگاریتمی در طول محور فرکانس پراکنده شده‌اند. هر فیلتر، انرژی یک باند فرکانسی خاص را جمع‌آوری کرده و به یک مقدار واحد تبدیل می‌کند. این فیلترها در فرکانس‌های پایین متراکم‌تر و در فرکانس‌های بالا پراکنده‌تر هستند که همان مقیاس مل را ایجاد می‌کنند.
* **تبدیل به مقیاس لگاریتمی :** در نهایت، مقادیر انرژی حاصل به مقیاس لگاریتمی (دسی‌بل) تبدیل می‌شوند تا تفاوت‌های دینامیکی در صدا (مانند صدای بلند و آرام) برجسته شوند و با نحوه‌ی درک انسان از بلندی صدا همسو گردند.

نتیجه نهایی این فرآیند، یک مل-اسپکتروگرام است که تصویری غنی از محتوای فرکانسی و زمانی قطعه موسیقی را در مقیاسی ادراکی ارائه می‌دهد و به عنوان ورودی ایده‌آل برای طبقه‌بندی سبک‌های موسیقایی عمل می‌کند. برای استخراج این ویژگی توسط پایتون می‌توان از کتابخانه‌ای به نام "Librosa" استفاده کرد Librosa .یک کتابخانه‌ی پایتون بسیار محبوب و کارآمد برای تحلیل موسیقی و سیگنال‌های صوتی است. این کتابخانه ابزارهای قدرتمندی برای انجام کارهای مختلفی از جمله بارگذاری داده‌های صوتی، پیش‌پردازش، استخراج ویژگی‌های صوتی و نمایش آن‌ها فراهم می‌کند.

در این پژوهش، از Librosa برای استخراج مل-اسپکتروگرام به عنوان یکی از ویژگی‌های کلیدی استفاده کردیم. این فرآیند چند مرحله‌ای، سیگنال صوتی را به یک تصویر بصری تبدیل می‌کند که نمایانگر محتوای فرکانسی در طول زمان در مقیاس ادراکی مل است. مراحل استخراج ویژگی با Librosa :

1. ابتدا باید توسط تابع‌های مربوطه، صوت را بارگذاری کنیم. این تابع فایل صوتی را به آرایه‌های Numpy (که همان سیگنال صوتی است) و نرخ نمونه‌برداری آن تبدیل می‌کند.
2. سپس، از تابع “librosa.feature.melspectrogram” برای استخراج مل-اسپکتروگرام استفاده می‌کنیم. این تابع به صورت خودکار مراحل تبدیل فوریه زمان کوتاه (STFT) و اعمال فیلتربانک مل را انجام می‌دهد.
3. melspectrogram معمولاً در مقیاس توان (power) است. برای نمایش بهتر و همسو شدن با درک انسانی از بلندی صدا، آن را با استفاده از تابع “librosa.power\_to\_db”به مقیاس لگاریتمی دسی‌بل (dB) تبدیل می‌کنیم.
4. در نهایت آرایه‌ی “mel\_spectrogram\_db”همان داده‌ی نهایی است که می‌تواند به عنوان ورودی به مدل‌های یادگیری عمیق، به‌ویژه شبکه‌های عصبی پیچشی (CNN)، داده شود تا ویژگی‌های سبک‌شناختی را از آن استخراج کنند.

این کتابخانه با ساده‌سازی این فرآیند، نقش حیاتی در آماده‌سازی داده‌ها برای تحلیل ما ایفا می‌کند.

#### ۳.۳.۲. تبدیل ویولت (Wavelet Transform)

تبدیل ویولت یک ابزار ریاضی قدرتمند برای تحلیل سیگنال است که رویکردی متفاوت از تبدیل‌های سنتی مانند تبدیل فوریه دارد. این تبدیل، به جای تحلیل سیگنال در یک بازه زمانی کلی، آن را در سطوح مختلف فرکانس و در زمان‌های متفاوت بررسی می‌کند. این ویژگی کلیدی، ویولت را برای تحلیل سیگنال‌های غیرپایدار مانند موسیقی که ویژگی‌های آن‌ها به صورت مداوم در حال تغییر است، بسیار کارآمد می‌سازد[[۴۲]](#link42).

ویولت در زبان ساده به معنای "موجک" است؛ یک موج با طول محدود و شکل مشخص که به عنوان یک تابع پایه برای تحلیل سیگنال اصلی به کار می‌رود. برخلاف تبدیل فوریه که از توابع سینوسی بی‌نهایت استفاده می‌کند، تبدیل ویولت با استفاده از این موجک‌های محدود، اطلاعات فرکانس و زمان را به صورت همزمان استخراج می‌کند[[۴۳]](#link43).

روند اصلی تبدیل ویولت به دو فرآیند کلیدی بستگی دارد :

* **جابجایی (Shifting):** موجک در طول سیگنال حرکت می‌کند تا در هر لحظه از زمان، همبستگی خود را با آن بخش از سیگنال محاسبه کند. این کار به ما می‌گوید که چه فرکانسی در چه زمانی رخ داده است.
* **کشیدگی/انقباض (Scaling):** موجک کشیده یا منقبض می‌شود تا بتواند فرکانس‌های مختلف را تحلیل کند. یک موجک کشیده (با فرکانس پایین) برای تحلیل تغییرات کلی و آهسته‌ی سیگنال (مانند هارمونی‌ها)، و یک موجک منقبض (با فرکانس بالا) برای تشخیص جزئیات و تغییرات سریع (مانند ریتم) استفاده می‌شود.

این ترکیب از جابجایی و کشیدگی، یک نقشه‌ی دقیق و دو بعدی از فرکانس و زمان فراهم می‌کند که به نقشه‌ی زمان-فرکانس معروف است.

چیزی که واضح است آن است که در نهایت تبدیل موجک نیز از دسته‌ی تبدیلاتی است که بر روی فرکانس اثرگذار است و ویژگی‌های آن نیز روی فرکانس دیده می‌شود. بنابراین برای اینکه در کنار تبدیل مل از آن استفاده کنیم، نیاز به دانستن برتری‌های ذاتی این تبدیل هستیم :

* **تفکیک زمان-فرکانس (Time-Frequency Resolution) :**
* **مل-اسپکتروگرام (STFT) :** در تبدیل فوریه (که پایه مل-اسپکتروگرام است)، تفکیک زمان و فرکانس به هم وابسته است. اگر بخواهیم فرکانس‌ها را با دقت بالا تحلیل کنیم، باید فریم‌های بزرگ‌تری انتخاب کنیم که باعث کاهش دقت زمانی می‌شود (یعنی نمی‌توانیم بگوییم یک نت دقیقاً در کدام لحظه نواخته شده است).
* **ویولت :** تبدیل ویولت این محدودیت را ندارد. در فرکانس‌های پایین، تفکیک فرکانسی بالایی دارد اما تفکیک زمانی کمتری، که برای تحلیل هارمونی‌ها و ساختار کلی موسیقی مناسب است. در فرکانس‌های بالا، تفکیک زمانی بسیار بالایی دارد اما تفکیک فرکانسی کمتری، که برای تشخیص جزئیات ریتمیک و تکنیک‌های سریع نوازندگی (مانند استکاتو و آرتیکولاسیون) بسیار کارآمد است. این ویژگی به ویولت اجازه می‌دهد تا تغییرات سریع را به صورت دقیق‌تری ثبت کند.
* **تحلیل غیرپایدار (Non-Stationary Signal Analysis):**
* مل-اسپکتروگرام: این روش برای تحلیل سیگنال‌های پایدار (ثابت) طراحی شده است و ممکن است در مواجهه با سیگنال‌های غیرپایدار مانند موسیقی که دائماً در حال تغییر هستند، عملکرد مطلوبی نداشته باشد.
* ویولت: این تبدیل به طور ذاتی برای تحلیل سیگنال‌های غیرپایدار بهینه‌سازی شده است. از آنجا که ویولت در طول زمان جابجا می‌شود، می‌تواند به بهترین شکل پدیده‌های گذرا و لحظه‌ای در موسیقی، مانند شروع و پایان یک نت، تغییرات ناگهانی در دینامیک یا تغییرات تمپو، را ثبت و تحلیل کند. این جزئیات دقیق، بخش جدایی‌ناپذیر از "اثر انگشت" یک نوازنده هستند.
* **تطابق با ویژگی‌های نوازندگی :**
* ویولت در تشخیص ویژگی‌های دقیق نوازندگی مانند "استکاتو در مقابل لگاتو"، "شدت ضربه بر کلاویه‌ها (velocity)" و "حالت‌های مختلف دینامیک" ، بسیار قوی عمل می‌کند. این ویژگی‌ها که اغلب در تحلیل‌های مبتنی بر مل-اسپکتروگرام گم می‌شوند، اطلاعات حیاتی برای تمایز میان سبک‌های مختلف یک نوازنده فراهم می‌کنند.

به همین دلیل، تبدیل ویولت یک ابزار مکمل و بسیار ارزشمند برای تحلیل‌های موسیقایی ماست. این ویژگی به همراه مل-اسپکتروگرام، به مدل ما یک دید جامع و چندبعدی از موسیقی می‌دهد که برای شناسایی دقیق "اثر انگشت" آهنگسازان و نوازندگان ضروری است.

از نظر تحلیل ریاضی باید بدانیم، تبدیل ویولت پیوسته (CWT) از نظر ریاضی به صورت زیر تعریف می‌شود:

که در آن :

* سیگنال اصلی در حوزه زمان است.
* *ψ(t)* تابع ویولت مادر (mother wavelet) است.
* ψ∗مزدوج مختلط تابع ویولت است.
* τ پارامتر جابجایی (translation) است که موقعیت ویولت در زمان را مشخص می‌کند.
* sپارامتر مقیاس (scale) است که کشیدگی یا انقباض ویولت را تعیین می‌کند.
* ∣s∣​عامل نرمال‌سازی انرژی است تا انرژی ویولت در هر مقیاس ثابت بماند.

این فرمول نشان می‌دهد که تبدیل ویولت چگونه سیگنال را به صورت انتگرال، با نسخه‌های جابجا و کشیده‌شده‌ی تابع ویولت، مقایسه می‌کند. هر مقدار در خروجی (Xω​) نشان‌دهنده‌ی میزان همبستگی بین سیگنال و ویولت در یک زمان و مقیاس خاص است[[۴۴]](#link44). تبدیل ویولت شامل کاربردهای عملی دیگری نیز می‌شود برای مثال :

1. **حذف نویز از سیگنال (Denoising):** یکی از کاربردهای مهم ویولت، حذف نویز از سیگنال‌هاست. فرض کنید یک سیگنال صوتی یا تصویری پر از نویزهای ناخواسته دارید. تبدیل ویولت می‌تواند سیگنال اصلی را از نویز جدا کند. این کار به این صورت انجام می‌شود که ابتدا سیگنال را به ضرایب ویولت تبدیل می‌کنیم. سپس، ضرایب کوچکی که معمولاً مربوط به نویز هستند، حذف می‌شوند . با تعیین آستانه(Thresholding) در نهایت، با استفاده از ضرایب باقی‌مانده، سیگنال را به حالت اصلی بازمی‌گردانیم. نتیجه، یک سیگنال تمیزتر و با کیفیت‌تر است. این تکنیک در پزشکی برای فیلتر کردن نویز از سیگنال‌های ECG و EEG، یا در نجوم برای حذف نویز از تصاویر تلسکوپ‌ها استفاده می‌شود.
2. **فشرده‌سازی داده‌ها (Data Compression):** ویولت در فشرده‌سازی داده‌ها، به‌ویژه تصاویر، کاربرد گسترده‌ای دارد. استاندارد فشرده‌سازی JPEG 2000 بر پایه تبدیل ویولت بنا شده است. علت این موضوع این است که ویولت می‌تواند اطلاعات مهم و اصلی یک تصویر را در تعداد کمی از ضرایب نگه دارد، در حالی که اطلاعات فرعی و کم‌اهمیت را با ضرایب بسیار کوچک نشان می‌دهد. با حذف این ضرایب کوچک، می‌توان حجم فایل را به شدت کاهش داد، بدون اینکه کیفیت تصویر به صورت محسوسی افت کند. این روش از فشرده‌سازی با تبدیل فوریه کارآمدتر است، زیرا ویولت قادر است به صورت بهتری از لبه‌ها و جزئیات مهم تصویر نگهداری کند.
3. **تشخیص ناهنجاری (Anomaly Detection):** ویولت در تشخیص ناهنجاری‌ها و تغییرات ناگهانی در داده‌ها بسیار قوی است. در بسیاری از سیستم‌ها، ناهنجاری‌ها به صورت سیگنال‌های گذرا و غیرمعمول ظاهر می‌شوند. از آنجایی که ویولت در تحلیل سیگنال‌های گذرا بسیار دقیق عمل می‌کند، می‌تواند برای شناسایی این ناهنجاری‌ها در حوزه‌های مختلف استفاده شود:
   * + **تحلیل زلزله :** برای تشخیص ریزش‌های کوچک در سیگنال‌های لرزه‌ای که ممکن است نشان‌دهنده یک زلزله قریب‌الوقوع باشند.
     + **مانیتورینگ ماشین‌آلات صنعتی :**  برای شناسایی لرزش‌های غیرعادی در یک موتور یا دستگاه که نشان‌دهنده نقص فنی است.
     + **تحلیل بازار مالی :** برای تشخیص جهش‌های ناگهانی و غیرمعمول در قیمت سهام یا ارزها که ممکن است به یک رویداد خاص مرتبط باشد.

در انتهای این توضیحات مشخص شد که تبدیل موجک علاوه بر ابزاری قدرتمند برای تحلیل‌های پردازش تصویر، می‌تواند تحلیلی دقیق از داده‌های موسیقی نیز ارائه دهد.

#### ۳.۳.۳. اسپکترال کانترست (Spectral Contrast)

اسپکترال کانترست یک ویژگی کم‌پارامتر است که تفاوت بین اوج‌های انرژی فرکانسی (peaks) و دره‌های انرژی فرکانسی (valleys) را اندازه‌گیری می‌کند. این ویژگی به ما کمک می‌کند تا کیفیت تُن (timbre) و بافت صوتی (texture) را تحلیل کنیم. بافت‌های غنی و پیچیده دارای اسپکترال کانترست بالایی هستند، در حالی که صداهای ساده‌تر این ویژگی را کمتر نشان می‌دهند. به دلیل پارامترهای کمتر، این ویژگی به مدل‌های ساده‌تر و سبک‌تر نیاز دارد و در عین حال نتایج مطلوبی ارائه می‌دهد. استفاده از این ویژگی به عنوان یک مکمل، به ما اجازه می‌دهد تا با صرف محاسبات کمتر، اطلاعات ارزشمندی را به مدل اضافه کنیم. برای درک بهتر، یک سیگنال صوتی را در نظر بگیرید که به طیف فرکانسی تبدیل شده است. این طیف، میزان انرژی را در هر فرکانس نشان می‌دهد. در این طیف، برخی فرکانس‌ها دارای انرژی بسیار بالا هستند (مانند نت‌های اصلی و هارمونیک‌ها) و برخی دیگر دارای انرژی بسیار پایین (مانند نویزها و فرکانس‌های بینابینی).

اسپکترال کانترست با شناسایی این اوج‌ها و دره‌ها، تفاوت بین آن‌ها را محاسبه می‌کند. یک قطعه موسیقی با بافت غنی و پیچیده (مثلاً یک قطعه ارکسترال با سازهای مختلف و هارمونی‌های پیچیده) دارای اوج‌ها و دره‌های فرکانسی زیادی است که منجر به کنتراست طیفی بالا می‌شود. در مقابل، یک سیگنال صوتی ساده (مانند یک صدای سینوسی یا یک نت تنها از یک ساز) دارای کنتراست طیفی پایین است، زیرا تفاوت بین اوج‌ها و دره‌ها کم است. این ویژگی شامل مزایایی است که استفاده از آن را بسیار کاربردی می‌کند :

 **تحلیل بافت صوتی :** این ویژگی به طور مستقیم با کیفیت تُن و بافت صوتی مرتبط است. بافت‌های غنی، پر از سازهای مختلف و هارمونی‌های متنوع، کنتراست طیفی بالایی دارند که این امر به مدل اجازه می‌دهد تا این تفاوت‌ها را به راحتی تشخیص دهد.

 **کم‌ پارامتر بودن :** اسپکترال کانترست یک ویژگی با پارامترهای نسبتاً کم است. این بدان معناست که برای پردازش آن به مدل‌های پیچیده و حجیم نیاز نداریم. یک مدل ساده‌تر و سبک‌تر نیز می‌تواند با استفاده از این ویژگی، نتایج مطلوبی به دست آورد که منجر به صرفه‌جویی در منابع محاسباتی و زمان آموزش می‌شود.

 **مکمل سایر ویژگی‌ها :** اسپکترال کانترست اطلاعاتی را ارائه می‌دهد که سایر ویژگی‌ها مانند مل-اسپکتروگرام یا تبدیل ویولت ممکن است به طور کامل پوشش ندهند. این ویژگی بر روی تفاوت‌های طیفی تمرکز دارد، در حالی که مل-اسپکتروگرام بر روی الگوهای فرکانسی و ویولت بر روی جزئیات زمانی و فرکانسی تمرکز می‌کند. اطلاعاتی که از هر سه این ویژگی‌ها بدست می‌آید می‌تواند به مدل یک درک جامع و چندبعدی از یک قطعه‌ی موسیقی بدهد.

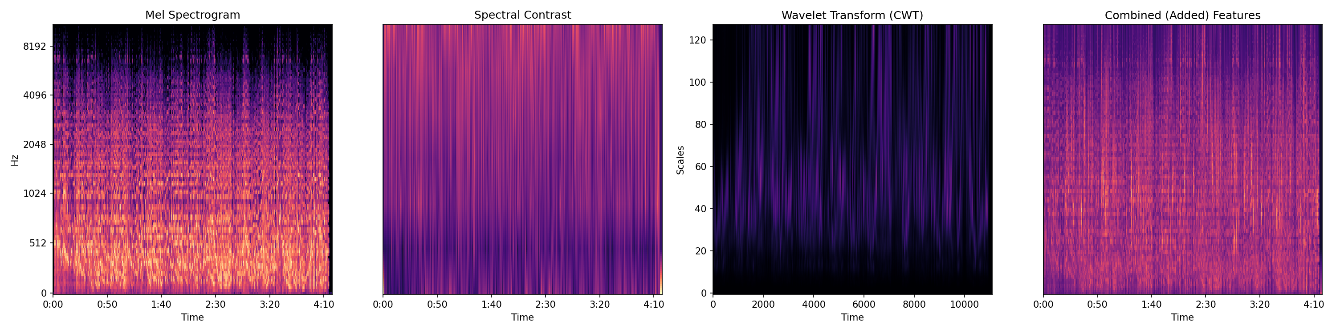
۳.۳.۴. ترکیب ویژگی‌ها

رویکرد ترکیبی برای استفاده از هر سه ویژگی به دو صورت امکان پذیر است :

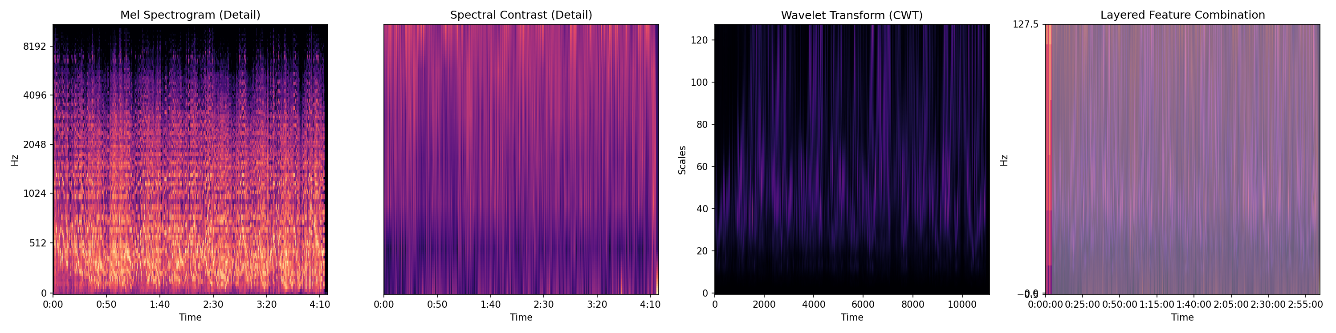
1. **ترکیب ویژگی‌ها قبل از یادگیری (Early Fusion):** در این رویکرد، ویژگی‌های استخراج‌ شده از هر قطعه موسیقی در یک بردار یا ماتریس واحد ادغام می‌شوند و سپس این ماتریس به عنوان ورودی واحد به مدل یادگیری عمیق داده می‌شود. این فرآیند به مدل اجازه می‌دهد تا از همان ابتدا، ارتباطات و وابستگی‌های بین ویژگی‌های مختلف را یاد بگیرد. ترکیب ویژگی‌ها می‌تواند با استفاده از روش‌های زیادی انجام شود، فرض کنید برای هر سگمنت از موسیقی، یک مل-اسپکتروگرام، یک ماتریس ویولت و یک بردار اسپکترال کانترست داریم. این سه ماتریس/بردار می‌توانند به صورت‌های مختلفی با هم ترکیب شوند، مثلاً از طریق کانکت‌ کردن (Concatenation) یا ترکیب در یک ماتریس چند کاناله (مانند یک تصویر رنگی که هر کانال آن یک ویژگی متفاوت است). بردارهای ویژگی ترکیبی می‌توانند با تاکید برروی شباهت‌ها، تفاوت‌ها و ویژگی‌های دیگر ساخته شوند. برای هر کدام از این موارد یک عمل ریاضی متفاوت برروی ماتریس ترکیب انجام می‌شود. برای مثال زمانی که بخواهیم تمامی اطلاعات را در بردار ترکیب شده داشته باشیم، ماتریس‌های مربوط به هر سه ویژگی را باهم جمع می‌زنیم. این نحو از یادگیری شامل مزایا و معایبی است که در ادامه بررسی خواهند شد :

* **مزایا :**
* **یادگیری جامع:** مدل از همان لایه‌های اول خود شروع به یادگیری وابستگی‌های بین ویژگی‌های مختلف می‌کند. این امر می‌تواند منجر به کشف الگوهای پیچیده‌تر و جامع‌تر شود.
* **ساده‌سازی معماری مدل :**  تنها به یک مدل یادگیری عمیق نیاز است که ورودی ترکیبی را می‌گیرد. این کار طراحی و مدیریت مدل را ساده‌تر می‌کند.
* **بهره‌وری محاسباتی :** از آنجایی که تنها یک فرآیند آموزش و یک مدل وجود دارد، ممکن است از نظر محاسباتی و زمانی کارآمدتر باشد.
* **معایب:**
* **افزایش ابعاد ورودی:** با ترکیب ویژگی‌ها، ابعاد ورودی به شدت افزایش می‌یابد. این امر می‌تواند منجر به مشکلاتی مانند نفرین ابعاد (Curse of Dimensionality) شود و نیاز به مدل‌های بزرگ‌تر و داده‌های بیشتر برای جلوگیری از بیش‌برازش (Overfitting) دارد.
* **ایجاد تداخل:** اگر ویژگی‌ها ماهیت بسیار متفاوتی داشته باشند (مثلاً یک ویژگی بسیار پرپارامتر و دیگری کم‌پارامتر)، ترکیب آن‌ها می‌تواند منجر به ایجاد تداخل شود و مدل نتواند به خوبی از ویژگی‌های کم‌اهمیت استفاده کند.

برای درک بهتر این موضوع نیاز است مثالی را برای ترکیب ویژگی‌ها مشاهده کنیم :



شکل ۶-۳ ویژگی‌ها و ترکیب آن‌ها به صورت تاکید بر روی تفاوت‌ها



شکل ۷-۳ ویژگی‌ها و ترکیب‌ آن‌ها با حفظ تمامی جزئیات

1. **استفاده از مفهوم آنسامبل** **(Late Fusion):** در این رویکرد، برای هر ویژگی یک مدل یادگیری عمیق جداگانه آموزش داده می‌شود. پس از آنکه هر مدل به صورت مستقل به پیش‌بینی خود رسید، نتایج آن‌ها با یکدیگر ترکیب می‌شوند تا یک نتیجه نهایی به دست آید. این ترکیب می‌تواند به روش‌های مختلفی مانند رأی‌گیری (Voting) یا میانگین‌گیری انجام شود. در فصل چهارم و در بخش اجرای مدل به نحوه‌ی اجرای این مفهوم در این پژوهش می‌پردازیم. برای استفاده‌ی درست از هر سه ویژگی در این روش نیاز است که مدل مخصوص به هر ویژگی استفاده شود تا مشخصات متمایز آن‌ها از دست نرود. هر دوی تبدیل ویولت و مل-اسپکتروگرام شامل پارامترهای زیادی هستند، بنابراین شبکه‌های عصبی پیچیده‌تر که بخصوص شامل وزن‌های پیشین برروی دیتاست‌های تصویر هستند برای آن‌ها مناسب تر هستند (در فصل چهارم به تفضیل شرح داده خواهد شد). برای یک مشخصه‌ی کم‌پارامتر مانند اسپکترال-کانترست نیز می‌توان از یک مدل پیچشی (CNN)، با لایه‌های کم استفاده کرد. اگرچه برای این ویژگی مدل باید شامل لایه‌های کم باشد، اما باید توجه داشت که ساختار لایه‌های کم باید برای استخراج ویژگی‌ها دارای پیچیدگی بیشتری باشند.

* **مزایا :**
* **تخصص‌گرایی مدل‌ها:** هر مدل به صورت تخصصی بر روی یک ویژگی خاص آموزش می‌بیند و بهترین عملکرد را از آن استخراج می‌کند. این رویکرد به مدل‌ها اجازه می‌دهد تا به خوبی با ماهیت متفاوت هر ویژگی سازگار شوند.
* **مقاومت در برابر خطا:** اگر یکی از مدل‌ها به دلایلی (مثلاً نویز زیاد در یک ویژگی) نتواند به درستی پیش‌بینی کند، مدل‌های دیگر می‌توانند این خطا را جبران کنند. این باعث افزایش مقاومت و ثبات سیستم می‌شود.
* **مدیریت آسان‌تر ابعاد :**  هر مدل با ورودی با ابعاد کمتر سروکار دارد، که مدیریت آن آسان‌تر است و از مشکلات ناشی از ابعاد بالا جلوگیری می‌کند.
* **معایب :**
* **پیچیدگی معماری:** طراحی، آموزش و مدیریت چندین مدل به صورت همزمان پیچیده‌تر است.
* **افزایش زمان و هزینه :**  آموزش چندین مدل به صورت جداگانه به زمان و منابع محاسباتی بیشتری نیاز دارد.
* **عدم یادگیری ارتباطات عمیق:** مدل‌ها به صورت مستقل یاد می‌گیرند و ممکن است نتوانند ارتباطات عمیق و وابستگی‌های پنهان بین ویژگی‌های مختلف را کشف کنند، مگر اینکه این ترکیب در لایه‌های عمیق‌تری از مدل صورت بگیرد.

#### ۳.۴. نتیجه‌گیری

این فصل، زیرساخت‌های لازم برای تحلیل و مدل‌سازی داده‌ها را فراهم کرد. با وجود چالش‌هایی مانند بالانس کردن دیتاسِت و مدیریت حافظه، توانستیم داده‌های خام را به مجموعه‌ای متعادل و قابل استفاده تبدیل کنیم. سپس، با استفاده از تکنیک‌های پیش‌پردازش، داده‌ها را برای استخراج ویژگی‌های کلیدی آماده کردیم. در نهایت، با استخراج ویژگی‌های مل اسپکتروگرام، تبدیل ویولت و اسپکترال کانترست، سیگنال‌های صوتی را به داده‌های قابل پردازش توسط مدل‌های یادگیری عمیق تبدیل کردیم. این فرآیند، زمینه‌ساز طراحی و آموزش مدل اصلی ماست که در فصل آینده به تفصیل به آن خواهیم پرداخت. در فصل بعدی، به سراغ معماری مدل‌های مبتنی بر پردازش تصویر و یادگیری عمیق خواهیم رفت که با استفاده از ویژگی‌های استخراج شده، قادر به شناسایی سبک نوازنده و تطبیق آن با آهنگسازان مختلف خواهند بود. این مدل‌ها به طور خاص از ویژگی‌های سیگنال پروسسینگ مانند تبدیل ویولت بهره خواهند برد تا به دقت مطلوب دست یابند.