Utilizando técnicas de visão computacional para reconhecimento de ações em vídeos de futebol

Gabriel Rocha Martins
Universidade Federal de Minas Gerais
Belo Horizonte, Brasil
garoma20@ufmg.br

Erickson Rangel do Nascimento
Universidade Federal de Minas Gerais
Belo Horizonte, Brasil
erickson@dcc.ufmg.br
Orientador

Abstract

Este relatório apresenta o progresso de um trabalho que propõe uma nova tarefa: detecção de tipos de eventos caracteríticos do futebol com base em vídeos. Como resultados preliminares do desenvolvimento do projeto, três contribuições se destacam: a definição da tarefa de classificação de eventos em vídeos de futebol baseada em um formato de dados de evento específico, a criação de um banco de dados composto de 55 vídeos de jogos completos dos quais existem dados de eventos anotados e a geração de modelos de visão computacional preliminares que classificam as ações dos vídeos em 2 diferentes escopos: uma ferramenta que classifica os vídeos dentre 10 classes com precisão média entre as classes de 38,34% e outra que classifica os vídeos dentre 36 classes, que são especializações das classes do modelo anterior, com precisão média entre as classes de 23,17%.

1. Introduction

O futebol é um esporte que surgiu na Inglaterra durante o século 19 e que hoje é jogado no mundo todo. Segundo relatório anual da FIFA (Federação Internacional de Futebol) [2] o esporte gerou mais de 480 milhões de doláres apenas no ano de 2024, sendo grande parte desta renda advinda dos direitos de transmissão dos campeonatos organizados e direitos sobre campanhas publicitárias. Sendo assim, apesar de ser, a princípio, uma atividade de entretenimento, dada a grande movimentaçao financeira gerada pelos jogos, o setor se tornou relevante no cenário internacional, o que fez com que a enfoque no mercado aumentasse à medida que empresas de diversos setores viram no entretenimento um meio de alavancar seus negócios com campanhas publicitárias e afins.

Dado o crescimento do esporte como um todo, a competitividade aumentou à medida que bons resultados poderiam resultar em grandes contratos tanto para os jogadores, quanto para os clubes. Neste contexto, a busca por maneiras eficientes de melhorar o desempenhos das equipes tornou o uso das imagens de transmissões de jogos fontes de dados que poderiam ser utilizados para geração de estatísticas, análises táticas, análises técnicas, entre outras tarefas que pudessem cooperar com o aumento da performance das equipes. Com isso, houve o surgimento de empresas como Wyscout e Statsbomb que tornaram a produção de dados esportivos o seu negócio, tendo como produto principal informações estatíticas sobre os jogos dos principais campeonatos. Dentre os dados que podem ser gerados para cada jogo, os dados de eventos são fontes relevantes para a produção de diversos tipos de análises tanto do comportamento conjunto das equipes, quanto do individual de cada jogador, dado que ele apresenta uma lista de informações, como tempo de início e tipo de ação, sobre cada um dos eventos que acontecem ao longo de um jogo.

Porém, de acordo com a Wyscout[5], empresa relevante no mercado de produção de dados, a maior parte do trabalho que envolve a produçao desses dados é realizada manualmente. Dado o interesse em gerar dados de qualidade sobre os jogos analisados, para cada jogo ao menos 3 analistas especializados em coleta de dados esportivos realizam as anotações dos eventos, sendo que dos 3, um deles é responsável por supervisionar a qualidade dos dados gerados como um todo e os outros 2 são responsáveis por anotar os eventos de cada um dos times em questão. Considerando que grandes campeonatos como a Premier League, liga nacional inglesa, tem 380 jogos por ano e que a produção de dados sobre cada um dos jogos é custosa tanto em termos financeiros, quanto em relação ao tempo, a produção desses dados é de alto valor, tornando o acesso à esses dados restrito à poucos clubes que tenham como arcar com esses custos, além de limitar pequisas que utilizem esses dados.

A automatização da produção de dados estatísticos por meio de análises de vídeos das transmissões dos jogos pode,

neste contexto, tornar-se uma ferramenta útil para auxiliar a produção de dados esportivos. O uso de vídeos para a detecção de ações é uma tarefa abordada por pesquisas recentes, dado que a criação de grandes base de dados de vídeos como Kinetics[4] possibiltou avanços na produção de modelos de reconhecimento de ações que se beneficiam de características extraídas por meio de redes neurais para geração de anotações sobre vídeos. Porém, no contexto de vídeos de futebol a tarefa pode ser desafiadora, dado que a ocorrência de eventos ao longo dos jogos é desbalanceada devido à natureza do esporte, o que faz com que, mesmo que o problema possa ser visto como um subproblema da tarefa de reconhecimento de ações em vídeos, existem particularidades que incentivam a produção de soluções específicas para este tipo de dado.

1.1. Contribuições

Sendo assim, as contribuições deste trabalho se resumem a três fatores principais:

(i) Geração de bancos de dados com vídeos de futebol com eventos devidamente anotados.

(ii) Definição da tarefa de reconhecimento de eventos em vídeos de futebol.

(iii) Estabelecimento de um *baseline* para a tarefa proposta, de forma a estabelecer uma base comparativa para projetos futuros relacionados a essa tarefa.

Além disso, vale ressaltar que um dos objetivos do projeto é criar um arcabouço para o tratamento da tarefa, com a criação de um repositório público que disponibilizará todas as ferramentas necessárias para a replicação dos experimentos realizados e, principalmente, para a criação de novas estratégias que visam solucionar o problema proposto considerando os dados coletados.

2. Trabalhos relacionados

Este trabalho se relaciona principalmente com os tópicos Reconhecimento de Ações, *Soccer Video Datatasets* e *Soccer Event Datasets*. Faremos, portanto, uma breve análise sobre os principais trabalhos atuais relacionados a cada um desses tópicos.

Reconhecimento de Ações. A tarefa de reconhecimento de ações é caracterizada pela detecção de ações em segmentos de vídeo considerando um leque de classes de ações propostas pelo ser humano. Ao longo do tempo, diversas abordagens foram exploradas para solucionar este tipo de problema, no contexto dos vídeos o direcionamento usual para a solução é a expansão de arquiteturas de redes neurais 2d para o 3d[1], que considera o eixo temporal adicionado pelos vídeos. Desta maneira, os modelos de rede neural gerados são capazes de extrair características tanto espaciais, referentes aos quadros dos vídeos, quanto temporais, relacionando cada quadro levando em consideração o eixo temporal.

Outra abordagem possível é o uso de *Transformers* [6] que, no contexto de computação visual, contrasta com os modelos CNN (Rede Neural Convolucional) que focam em extrair características locais, ao propor a extração de características gerais dos dados de entrada, tornando o modelo mais robusto ao lidar com entradas de tamanhos maiores, o que é importante no contexto de dados de vídeo que adicionam o eixo temporal às imagens. Enfim, esses modelos agragam metodologias adotadas por modelos de reconhecimento de imagens, que recorrem à extração de características locais, com características de processamento de linguagem natural ao tentar obter informações utilizando o fator sequencial dos quadros.

Soccer Video Datasets. Tratando da geração de grandes bancos de dados de vídeos de futebol o trabalho que propõe *SoccerNet* [3] ganha destaque. Ao longo do desenvolvimento deste trabalho os autores foram capazes de gerar um banco de dados significativo com imagens de 500 jogos completos das seis principais ligas europeias durante três temporadas, de 2014 a 2017, resultando em um total de 764 horas de vídeo. Além dos vídeos, o banco de dados gerado também contém informações sobre três ações relacionadas aos jogos: gols, cartões e substituições, que foram obtidos extraindo informações dos relatórios dos jogos disponibilizados pelos sites das ligas em questão. Além disso, o trabalho propõe algumas tarefas possibilitadas pelos dados gerados, sendo uma delas a de detecção de ações nos vídeos, que propõe o uso de modelos de classificação de vídeos utilizando segmentos de vídeo gerados por uma janela deslizante de 1 segundo como entrada para gerar anotações sobre os possíveis eventos relacionados ao vídeo, uma tarefa que se relaciona com uma das propostas deste trabalho que é geração de segmentos de vídeo referentes a eventos de jogos de futebol.

Soccer Event Datasets. Já em relação à geração de dados de eventos associados à jogos de futebol o projeto em enfoque é o que propõe o banco de dados *Wyscout Top 5* [5]. Neste projeto, em associação com a empresa Wyscout, que tem a produção de dados esportivos como seu negócio, os autores desenvolveram um banco de dados que contém dados de eventos de todos os jogos da temporada 2017-2018 das cinco principais ligas da europa, além dos jogos da Eurocopa de 2016 e da Copa do Mundo de 2018. Sendo assim, como resultado, tornaram público o acesso a dados que contém informações relevantes sobre cada um dos eventos que ocorreram em cada um dos jogos em questão. Além disso, apresentam a metodologia de coleta desses dados que se apresentava como uma tarefa manual auxiliada por um software proprietário da empresa Wyscout. Ao longo do desenvolvimento do projeto há a apresentação de cenários de uso desse tipo de dado que sustentam a contribuição dele com os profissionais de análise esportiva.

3. Métodos

Este projeto propõe a solução de 3 tarefas principais: a criação de um banco de dados que relacione eventos de fuitebol à vídeos, a proposição de uma metodologia de segmentação dos vídeos baseada em dados de eventos e a adaptação de uma rede neural pré-treinada para a classificação dos segmentos gerados dentre classes de eventos pré-determinadas.

3.1. Geração do banco de dados

Dado que os dados de interesse eram vídeos com ações anotadas a metodolgia adotada foi utilizar as anotações disponíveis por dados de evento, que contém informações sobre diversos eventos que podem acontecer durante um jogo de futebol, para segmentar vídeos de jogos completos de futebol, de forma a criar de forma automática vídeos dos quais já houvessem anotações relativas às ações. Sendo assim, esta tarefa se divide em três outras tarefas: coleta de dados de evento, coleta de dados de vídeo e segmentação automática dos vídeos.

Coleta de dados de evento: Os dados de evento utilizados para o desenvolvimento do projeto são os do banco de dados *Wyscout Top 5* [5] que foram disponibilizados publicamente pelos autores por meio da plataforma *Figshare*. Ele contém dados de eventos de todos os jogos da temporada 2017-2018 das 5 principais ligas europeias: *La Liga*, *Premier League*, *Bundesliga*, *League 1* e *Serie A*, além dos jogos da Eurocopa de 2016, competição entre seleções europeias, e jogos da Copa do Mundo de 2018, maior competição de seleções do esporte. Com isso, estão disponíveis informações sobre 1941 jogos, que no total somam mais de 3 milhões de eventos anotados realizados por quase 4300 jogadores distintos. Sobre cada um dos eventos que ocorreram nesse jogos as seguintes informações estão disponíveis:

1. **eventId:** Um identificador do tipo de ação que ocorre no evento.
2. **eventName:** Nome do tipo de evento que ocorre no evento associado ao eventId.
3. **subEventId:** Um identificador do subtipo de ação que ocorre no evento.
4. **subEventName:** Nome do subtipo de ação que ocorre no evento associado ao subEventId.
5. **tags:** Uma lista que contém informações adicionais sobre o evento, utilizada para especificar alguns tipos de ações que estão subespecificados ainda pelas informações anteriores.
6. **eventSec:** O tempo de início do evento, considerando o tempo em segundos desde o início do tempo atual do jogo.
7. **id:** Um identificador único do evento.
8. **matchId:** Um identificador que referencia o jogo no qual o evento ocorreu.
9. **matchPeriod:** Um identificador do período do jogo em que o evento ocorreu, isto é, identifica se o evento ocorreu no primeiro ou segundo tempo do tempo normal ou prorrogação, ou, por fim, durante os pênaltis.
10. **playerId:** Um identificador do jogador que gerou o evento.
11. **positions:** Uma lista de tuplas contendo as informações sobre as posições de início e término do evento.
12. **teamId:** Um identificador do time do jogador que gerou o evento.

Coleta dos vídeos: Dado que o interesse era gerar vídeos para os quais já existiam dados de eventos anotados, a coleta dos vídeos foi direcionada à busca por vídeos de jogos completos dos quais havia referência nos dados de eventos coletados, ou seja, dos jogos da temporada 2017-2018 das cinco principais ligas europeias, da Eurocopa de 2016 e da Copa do Mundo de 2018. Para isso, foi utilizada a plataforma *Youtube* para realizar a busca, dado que os vídeos listados na mesma são públicos e que os servidores que mantém a aplicação são de grande capacidade, de forma a possibilitar a aquisição de vídeos em tempo hábil para a execução do projeto. Sendo assim, utilizamos os canais oficias das ligas em questão, além dos canais dos times que participam de cada uma dessas ligas, do canal oficial da Fifa e dos canais referentes às seleções que participaram dos campeonatos internacionais, para procurar por vídeos de jogos completos considerando a limitação inicial proporcionada pelo dados de evento. O procedimento utilizado para a procura era simples, por meio da ferramenta de busca interna da plataforma procuramos por vídeos com espressões chaves que remetessem à vídeos de jogos completos como, por exemplo, *'Full Match'* e após realizar as buscas manualmente em cada um dos canais oficiais que teriam potencialmente conteúdos de interesse adquirimos os arquivos referentes aos vídeos por meio de uma ferramenta chamada *Youtube DLP*, uma aplicação pública que possibilita a aquisição de arquivos de diversos sites como o *Youtube*.

Por fim, foi possível coletar vídeos referentes a 55 jogos completos que tem referências aos dados de evento, todos eles com resolução 1920 x 1080 ou 1280 x 720, considerando a limitação dos dados disponíveis e a quantidade de memória que poderia ser alocada para o armazenamento desses arquivos, dado que cada vídeo coletado ocupava entre 1 e 4 Gb de memória devido ao comprimento. Sendo assim, os vídeos coletados são referentes a apenas 4 dos 7 campeonatos dos quais existiam dados de eventos coletados, sendo que grande deles referentes a jogos da La Liga como pode ser visto na tabela 1.

Segmentação automática dos vídeos: Para segmentar os vídeos de forma automática eram necessários os tempos de início e fim de cada um do eventos de interesse nos vídeos dos jogos. Para isso, anotamos manualmente quais

Liga	Quantidade de jogos
Copa do Mundo	6
La Liga	42
Premier League	3
Serie A	4
Total	55

Table 1. Esta tabela mostra a distribuição dos vídeos coletados para cada um dos campeonatos dos quais ao menos um vídeo de jogo completo foi coletado.

eram os identificadores de jogos *'matchId'* para os quais haviam vídeos coletados. Feito isso, filtramos do banco de dados apenas os eventos referentes à esses jogos. Com isso, tínhamos o tempo de início de cada um dos eventos dos jogos coletados e o tipo de evento que ocorreu que também seria utilizado posteriormente para o treinamento do modelo. A próxima tarefa foi definir os tempos de fim de cada um dos eventos, para isso utilizamos o tempo de início do evento subsequente como tempo de fim para o atual, desta forma definimos tempos de início e fim para todos os eventos de interesse com excessão dos último eventos de cada tempo dos jogos, dado que para esse eventos não havia evento subsequente que permitisse a definição do tempo de fim. Com os tempos de início (tie) e fim (tfe) de cada evento em mãos agora era necessário relacionar esses tempos com o tempo dos vídeos, para isso utilizamos do fato de que a partir do início de um período de um jogo de futebol o cronômetro não para em nenhum momento, ou seja, há uma continuidade no tempo considerando cada período do jogo. Sendo assim, bastava determinar o tempo no vídeo em que o primeiro evento de cada um dos períodos do jogo ocorriam para determinar os tempos de início e fim de todos eventos daquele período utilizando os dados de eventos.

Considerando que os vídeos eram referentes a transmissões completas desses jogos, haviam imagens de conteúdos que não fossem exatamente do jogo como programas pré-jogo e de intervalo.Portanto, foi necessário determinar manualmente o tempo de início cada um dos períodos nos vídeos (tip) de forma que o tempo de início em vídeo (tiev) e tempo de fim em vídeo (tfev) de um evento pudesse ser calculado automaticamente com as expressões **tiev = tip + tie** e **tfev = tip + tfe**. Definidos os tempos necessários, o próximo passo era filtrar os dados de eventos de interesse para o desenvolvimento do projeto, para isso, duas metodologias foram utilizadas: utilizar a coluna *eventName* para classificar os eventos dentre 10 classes de eventos ou utilizar a coluna *subEventName* para classificar os eventos dentre 36 classes que representam especializações das 10 classes da coluna *eventName*.

Como resultado do processo de segmentação dos foram gerados aproximadamente 92 mil índices que poderiam ser utilizados para gerar vídeos curtos para cada um dos eventos contemplados pelos dados. Porém, devido à limitação de armazenamento do ambiente de desenvolvimento e ao tamanho dos dados gerados os vídeos não foram explicitamente gerados, os índices foram utilizados para pivotear o conteúdo de cada evento que geraria os dados que seriam utilizados pelos modelos de rede neural, a metodologia para o processamento dos dados será detalhada na sessão de experimentos4.

3.2. Reconhecimento de eventos

Considerando o objetivo do projeto a abordagem escolhida foi utilizar uma rede neural convolucional tridimensional para classificação. Neste contexto, a arquitetura escolhida para este processo foi a X3D-S [1] que apresenta uma proposta de expansão de CNN's bidimensionais simples ao longo de eixos dos vídeos como o eixo do tempo e a profundidade dos canais. O motivo pela opção é o custo benefício da proposta que, por expandir redes simples gradualmente pelo eixos do vídeo, geram resultados competitivos com outras abordagens que representam o estado da arte, sem apresentar o custo computacional alto que é um problema recorrente neste contexto. Além disso, dado a quantidade de dados gerados pela etapa anterior e que a arquitetura possui uma grande quantidade de parâmetros, por volta de 3.9 milhões, optou-se pela realização de um *fine tunneling* de um modelo pré-treinado em um banco de dados suficientemente grande para adequar todos os pesos do modelo. Sendo assim, o modelo escolhido para a realização da tarefa foi o modelo pré-treinado no banco de dados Kinetics 400 [4] conhecido no contexto de reconhecimento de ações, que é disponibilizado em uma biblioteca da linguagem *python* chamada *pytorchvideo*.

Dada a natureza do problema a ser resolvido, foi necessário adequar as camadas finais do modelo, que correspondiam ao classificador de fato. Para isso, inicialmente um modelo correspondente à arquitetura proposta foi gerado utilizando a função disponibilizada pela biblioteca *torchvision* e os mesmos hiperparâmetros do modelo pré-treinado. Feito isso, carregamos os pesos referentes às camadas iniciais do modelo que realizam a extração das caracteríticas e não estavam relacionadas diretamente com a tarefa de classificação. Com isso, o modelo gerado para a realização dos experimentos possuía os pesos obtidos com o treinamento no banco de dados *Kinetics* que eram relevantes para a extração de características e pesos aleatórios nas camadas que realizariam de fato a classificação, que foram adaptadas de acordo com o número de classes desejadas na saída para cada um dos experimentos realizados.

4. Experimentos

O modelo utilizado para os experimentos apresentados pelas seção anterior 3 recebe como entrada tensores de di-

mensões 13 x 3 x 244 x 244, que representam, respectivamente, o número de quadros, o número de canais e a resolução do vídeo. Sendo assim, para cada um dos eventos do banco de dados produzido era necessário gerar um tensor com as dimensões adequadas para o modelo. Com isso, a primeira tarefa era amostrar quadros dos eventos utilizando os índices gerados ao longo do processo de segmentação, dado que cada vídeo possuía diferentes comprimentos e o modelo tem entrada de tamanho fixo de 13 quadros. Para isso, foi realizada uma amostragem temporal uniforme dos quadros respectivos à cada evento para gerar os 13 quadros, ou seja, para cada vídeo calculava-se a quantidade total de quadros, dividia pelo número de quadros desejados (13) gerando a frequência com que os quadros deveriam ser amostrados de forma a respeitar o aspecto sequencial dos vídeos, isto é, para um vídeo com 130 quadros um quadro seria selecionado a cada 10 quadros. Feito isso, a segunda etapa era redimensionar os quadros para a resolução suportada pelo modelo de 244 x 244, o que foi feito utilizando a função *Resize* da biblioteca *python torchvision*. Considerando a quantidade de eventos e as limitações do ambiente de desenvolvimento no que se diz respeito ao armazenamento, para cada evento foi salvo um tensor com os dados brutos de cada um dos tensores que representavam um evento, ou seja, um tensor de 1 byte por célula e dimensões 13 x 3 x 252 x 252. Por fim, para a realização do treinamento era necessário normalizar os dados considerando a mesma média e desvio padrão utilizados para normalizar os dados que foram utilizados durante o pré-treinamento, que eram respectivamente de 0.45 e 0.225, para isso primeiramente os valores dos tensores foram escalados para o intervalo entre 0 e 1, em seguida a função *Normalize* foi utilizada para realizar a normalização nos quatro eixos do tensor. Por fim, para cada evento foram gerados tensores adequados para o modelo.

Características do treinamento: Com os dados adequados em mãos o próximo passo era realizar o treinamento do modelo. Para isso, foi utilizada a Entropia Cruzada Binária como função de perda que direcionaria os ajustes dos pesos do modelo aos dados, dado que é uma função comum e adequada para a tarefa em questão. Para adequar a taxa de aprendizado do modelo à medida que as épocas avançavam foi utilizado o otimizador Adam, que utiliza um janela de contexto dos resultados obtidos em cada época do treinamento para ajustar a taxa de aprendizado de maneira adequada aos resultados imediatos a cada época. A taxa inicial de aprendizado utilizada foi de 0.0001, um valor baixo dado que o modelo utilizado já havia passado por um processo de treinamento em um banco de dados grande e generalista, o que fez com que fosse considerado que o modelo com os pesos inciais seria capaz de extrair características suficientemente generalistas dos vídeos e que o processo de treinamento tinha como objetivo refinar esses pesos considerando o contexto específico dos dados do projeto em questão. Além disso, foi utilizada a técnica de *early stopping* para que passadas três épocas sem ganhos de desempenho do modelo durante o treinamento, considerando o valor da precisão média entre as classes nos dados de validação como métrica, o processo de treinamento era encerrado.

Treinamentos realizados: Até o presente momento, dois experimentos foram realizados com o intuito de compreender melhor o comportamento e desempenho do modelo em relação à tarefa. O primeiro experimento foi realizado utilizando a coluna *eventName* para a classificação dos eventos, que, após 8 épocas de treinamento (considerando o *early stopping* na época 10), convergiu para um modelo que distinguia os eventos entre 10 classes com precisão média entre elas de 38,34%. Já no segundo experimento realizado a coluna *suvEventName* foi utilizada para classificar os eventos dentre 36 classes que eram especializções das classes do modelo anterior, que após 5 épocas (considerando o *early stopping* na época 7) convergiu para uma precisão média de 23,17%.

5. Conclusão

Dados os resultados encontrados até o momento é perceptível que os modelos gerados não foram capazes de capturar as características específicas de cada classe que poderiam distinguir dentre os eventos alvo. Analisando os segmentos de vídeo gerados pela metologia de segmentação adotada ao longo do desenvolvimento do projeto é possível notar que a mesma não foi capaz de segmentar os vídeos de maneira coerente para algumas das classes, dado que em alguns casos os eventos alvos se deslocavam arbitrariamente do evento alvo em questão, além de ignorar segmentos grandes gerados por momentos de replay ou bola parada caracteríticos de transmissões esportivas.

Outro fator importante revelado pelos experimentos é que apesar de cada subevento determinado pela coluna *subEventName* ter caracteríticas inerentemente diferentes o modelo com mais classes apresentou um desempenho médio pior, o que acredita-se ter sido causado por 2 motivos principais: segmentação falha especificamente para alguns dos subeventos o que pode ter como causa a qualidade dos dados fonte de eventos do dataset da *Wyscout* e a distribuição signicativamente desigual entre a quantidade de instâncias de cada um dos subveventos, que é gerada naturalmente pelo esporte mas que impacta diretamente no processo de treinamento de ferramentas como a utilizada.

Por fim, dentre os possíveis passos futuros dois deles se destacam em prol da melhora dos resultados obtidos: a adoção de novas estratégias de segmentação que adequem para cada tipo de subevento heurísticas que possam melhorar a representatividade de seus respectivos dados e criação de um subconjunto artificial dos eventos de forma a equili-

brar a quantidade de instâncias entre cada uma das classes. Além disso, caso nenhuma das estratégias apresentem resultados satisfatórios é possível realizar a segmentação manual de um subconjunto reduzido dos dados, de forma a garantir a qualidade dos dados a serem utilizados pelo modelo em detrimento da capacidade de generalização do mesmo.

References

[1] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 203–213, 2020. 2, 4

[2] Fifa. Fifa annual report 2024, 2024. 1

[3] Silvio Giancola, Mohieddine Amine, Tarek Dghaily, and Bernard Ghanem. Soccernet: A scalable dataset for action spotting in soccer videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1711–1721, 2018. 2

[4] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 2, 4

[5] Luca Pappalardo, Paolo Cintia, Alessio Rossi, Emanuele Massucco, Paolo Ferragina, Dino Pedreschi, and Fosca Giannotti. A public data set of spatio-temporal match events in soccer competitions. *Scientific data*, 6(1):236, 2019. 1, 2, 3

[6] Javier Selva, Anders S Johansen, Sergio Escalera, Kamal Nasrollahi, Thomas B Moeslund, and Albert Clapés. Video transformers: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):12922–12943, 2023. 2